

# AP16174

## XE164

UConnect-CAN XE164 "Cookery Book" for a  
hello world application

Using DAvE (Code Generator)

Using the KEIL tool chain µVision 4 (IDE,  
Compiler, Utility Tools)

Microcontrollers



Never stop thinking

**Edition 2010-04-22**

**Published by**

**Infineon Technologies AG  
81726 München, Germany**

**© Infineon Technologies AG 2010.  
All Rights Reserved.**

**LEGAL DISCLAIMER**

THE INFORMATION GIVEN IN THIS APPLICATION NOTE IS GIVEN AS A HINT FOR THE IMPLEMENTATION OF THE INFINEON TECHNOLOGIES COMPONENT ONLY AND SHALL NOT BE REGARDED AS ANY DESCRIPTION OR WARRANTY OF A CERTAIN FUNCTIONALITY, CONDITION OR QUALITY OF THE INFINEON TECHNOLOGIES COMPONENT. THE RECIPIENT OF THIS APPLICATION NOTE MUST VERIFY ANY FUNCTION DESCRIBED HEREIN IN THE REAL APPLICATION. INFINEON TECHNOLOGIES HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND LIABILITIES OF ANY KIND (INCLUDING WITHOUT LIMITATION WARRANTIES OF NON-INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS OF ANY THIRD PARTY) WITH RESPECT TO ANY AND ALL INFORMATION GIVEN IN THIS APPLICATION NOTE.

**Information**

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office ([www.infineon.com](http://www.infineon.com)).

**Warnings**

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

---

**AP08048****Revision History:** 2010-04 V2.0

Previous Version: none

Page	Subjects (major changes since last revision)

---

**We Listen to Your Comments**

Any information within this document that you feel is wrong, unclear or missing at all?

Your feedback will help us to continuously improve the quality of this document.

Please send your proposal (including a reference to this document) to:

[mcdocu.comments@infineon.com](mailto:mcdocu.comments@infineon.com)



---

Table of Contents	Page
-------------------	------

Note: Table of Contents [see page 9](#).

### Introduction:

This "Appnote" is a Hands-On-Training / Cookery Book / step-by-step book.  
It will help inexperienced users to get an UConnect-CAN XE164 up and running.

With this step-by-step book you should be able to get your first useful program in less than 2 hours.

The purpose of this document is to gain know-how of the microcontroller and the tool-chain.  
Additionally, the "hello-world-example" can easily be expanded to suit your needs.  
You can connect either a part of - or your entire application to the UConnect-CAN XE164.  
You are also able to benchmark any of your algorithms to find out if the selected microcontroller  
fulfils all the required functions within the time frame needed.

### **Note:**

The style used in this document focuses on working through this material as fast and easily as possible. That means there are full screenshots instead of dialog-window-screenshots; extensive use of colours and page breaks; and listed source-code is not formatted to ease copy & paste.

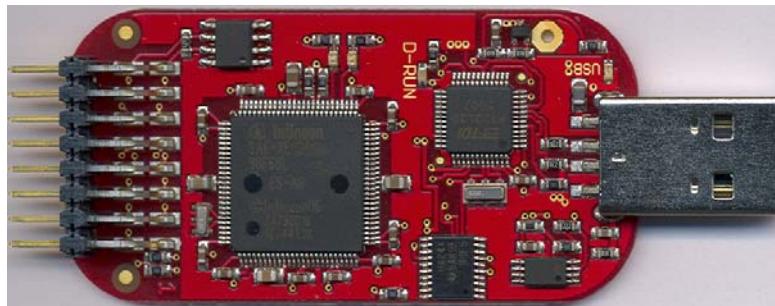
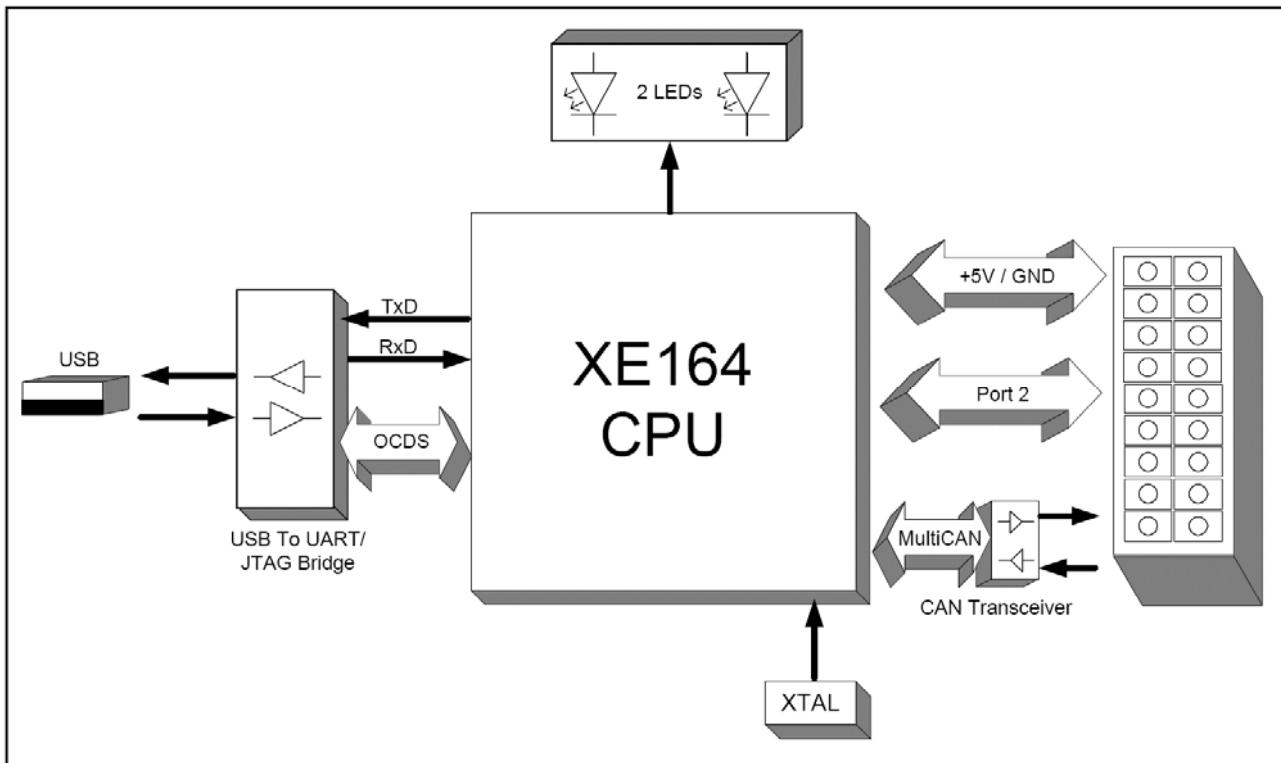
Have fun and enjoy the UConnect-CAN XE164!



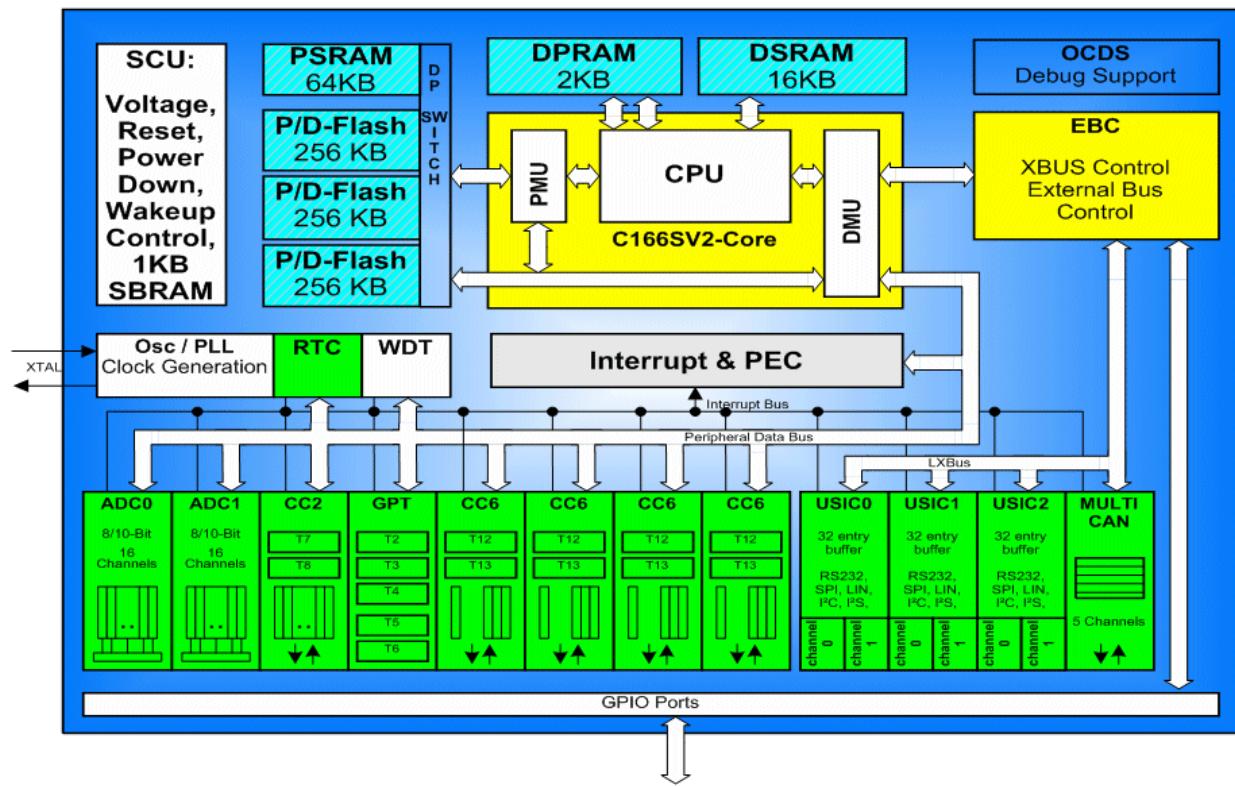
# Programming Example

## UConnect-CAN XE164

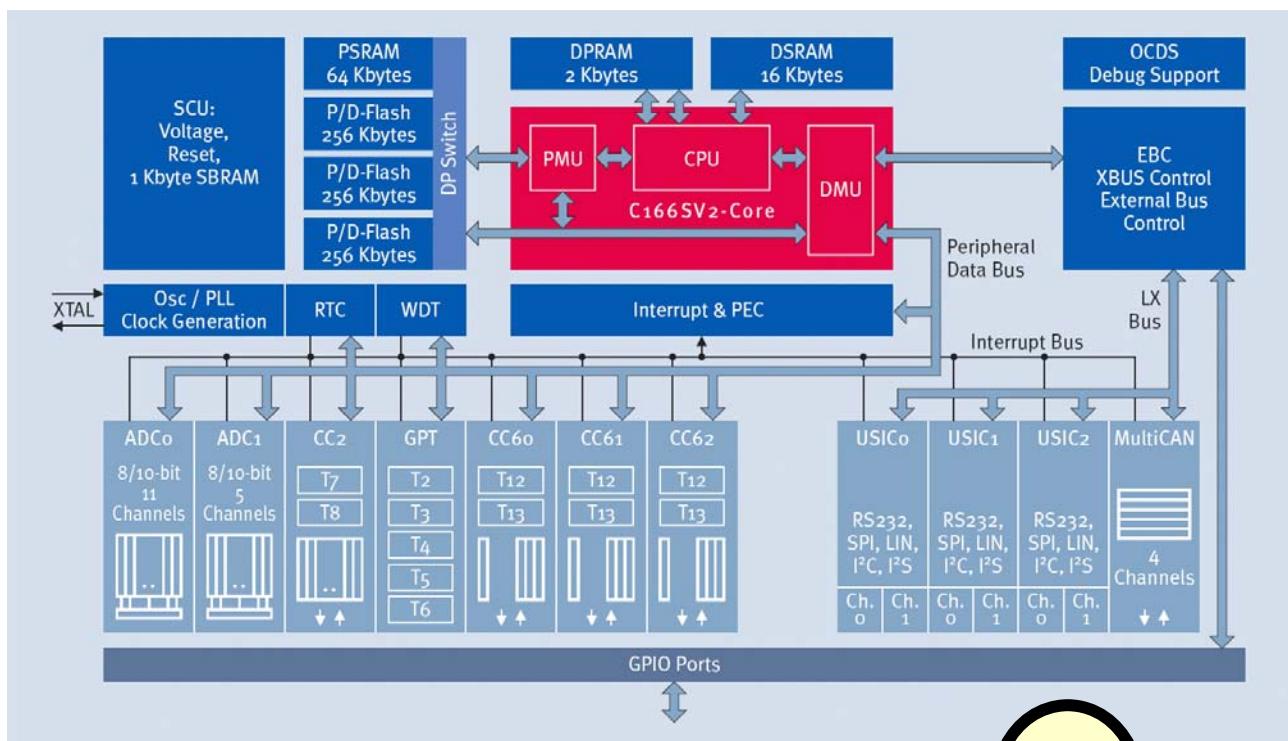


**Block Diagram** (Source: XE164 UConnect Manual)

## SAF-XE167F-96F66L Block Diagram (Source: Product Marketing)



## SAF-XE164F-96F66L Block Diagram (Source: Product Brief)

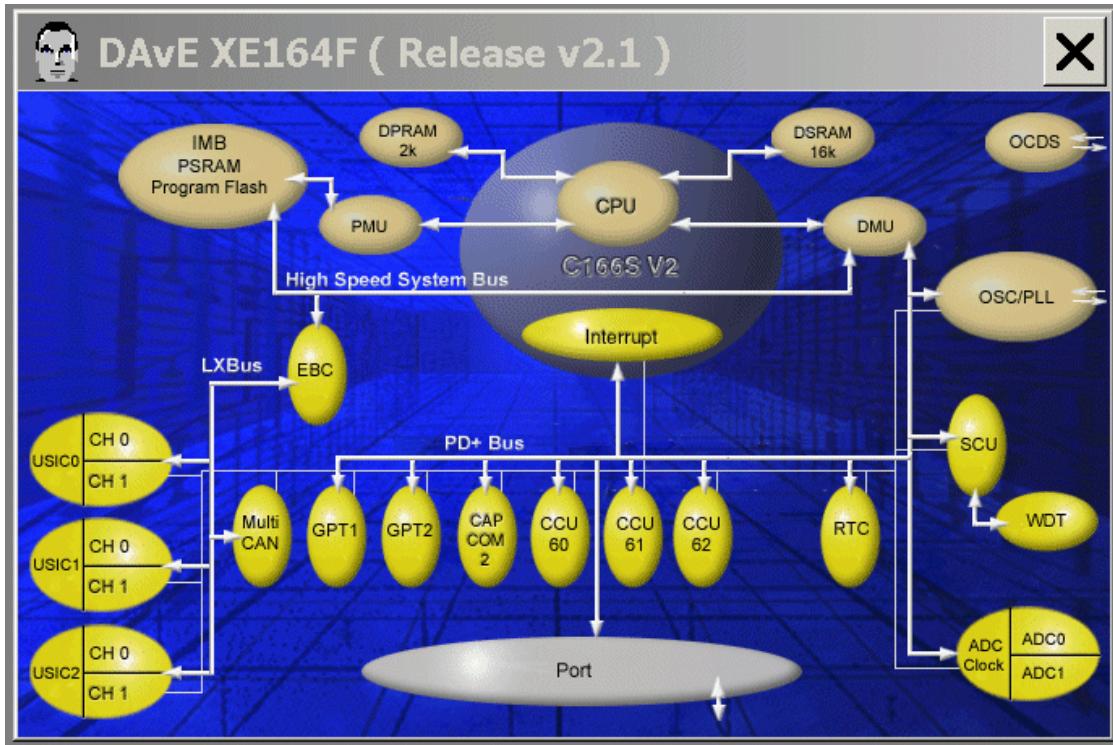


Note:

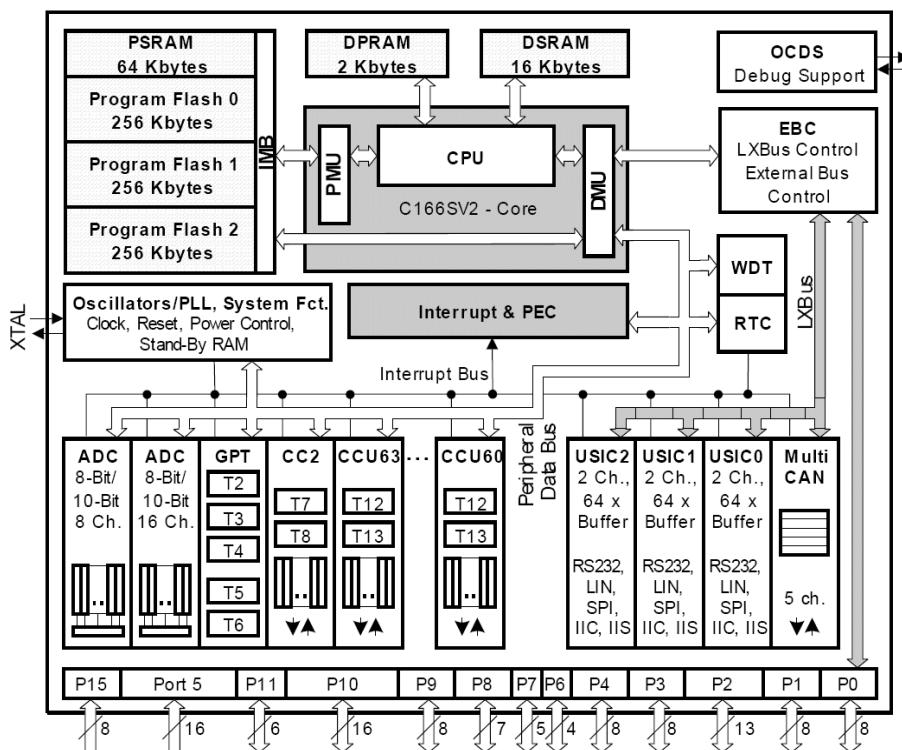
The XE164 microcontroller is a derivative of the XE167 microcontroller!



## SAF-XE164F-96F66L Block Diagram (Source: DAvE)



## XE16x Block Diagram (Source: User's Manual)



## Note:

Just by comparing the different sources of block diagrams, you should be able to get a complete picture of the microcontroller and to answer some of your initial questions.



**“Cookery book”**

For your first programming example for the UConnect-CAN XE164:

Your program:	<p>The screenshot shows the Docklight V1.7 interface. In the 'Send Sequences' tab, there is a list of three items:</p> <table border="1"> <thead> <tr> <th>Send</th> <th>Name</th> <th>Sequence</th> </tr> </thead> <tbody> <tr> <td>---&gt;</td> <td>1 ... LED IO_Port_2.7 ON</td> <td>1</td> </tr> <tr> <td>---&gt;</td> <td>2 ... LED IO_Port_2.7 OFF</td> <td>2</td> </tr> <tr> <td>---&gt;</td> <td>3 ... LED IO_Port_2.7 blinking</td> <td>3</td> </tr> </tbody> </table> <p>In the 'Communication' tab, the ASCII panel displays the following text:</p> <pre>1 ... LED IO_Port_2.7 ON 2 ... LED IO_Port_2.7 OFF 3 ... LED IO_Port_2.7 blinking</pre> <p>The 'your choice:' prompt is visible at the bottom of the communication window.</p>	Send	Name	Sequence	--->	1 ... LED IO_Port_2.7 ON	1	--->	2 ... LED IO_Port_2.7 OFF	2	--->	3 ... LED IO_Port_2.7 blinking	3
Send	Name	Sequence											
--->	1 ... LED IO_Port_2.7 ON	1											
--->	2 ... LED IO_Port_2.7 OFF	2											
--->	3 ... LED IO_Port_2.7 blinking	3											
Chapter/Step	<b>*** Recipes ***</b>												
1.)	<a href="#"><b>DAS Installation + Connecting the UConnect-CAN XE164</b></a>												
2.)	<a href="#"><b>DAvE (program generator)</b></a> <a href="#"><b>DAvE Installation (mothersystem) + DAvE Update Installation (XE16xx_Series.dip) for XE164</b></a>												
3.)	<a href="#"><b>Using DAvE</b></a> <a href="#"><b>Microcontroller initialization for your programming example</b></a>												
4.)	<a href="#"><b>Using the KEIL Development Tools (μVision4; C166 Version 7.00)</b></a> <a href="#"><b>Programming of your application (hello world) with the KEIL tool chain (μVision4)</b></a>												
5.)	<a href="#"><b>Running your first programming example</b></a>												
Feedback													
6.)	<a href="#"><b>Feedback</b></a>												

### 1.) DAS Installation + Connecting the UConnect-CAN XE164:



Screenshot of the UConnect-CAN XE164 Homepage:

<http://www.infineon.com/cms/en/product/channel.html?channel=db3a3043183a955501183c3446e1006f>

The screenshot shows the Infineon UConnect-CAN XE164 product page. At the top, there's a navigation bar with links for File, Edit, View, Favorites, Tools, Help, and a search bar. Below the header, the main content area has a breadcrumb trail: Home > Microcontrollers > Development Tools, Software and Kits > XE166 Development Tools and Software > Easy Kits, Evaluation Kits and Application Kits > UConnect-CAN XE164. The main title is "UConnect-CAN XE164". A brief description states: "The UConnect XE164 is a low cost USB stick providing full evaluation capability for the new XE166 16bit Family of microcontrollers. The kit includes development toolchains, demos, a CANopen EVA version and tutorials for quick installation and ease of use. The Uconnect USB Stick comes with an CAN extension Board." Below this, there's a section titled "UConnect XE164 Features" with a bulleted list of benefits. Another section, "CANopen EVA Version", lists software components like CANopen EVA version and Stick PC GUI. To the right, there's a large image of the UConnect stick and extension board, along with links to print and send the page. A "Related Links" section includes a link to the "UConnect OLED Extension Board". At the bottom, there's a "Buy Online" button, a copyright notice, and a "Trusted sites" badge.

**Note:**

For further information, please refer to the [XE164 UConnect Manual, V.1.0](#).  
For further information, please refer to the [XE164 UConnect Manual, V.1.1](#).





**Note:**

For the UConnect-CAN XE164 there is an additional extension board UConnect OLED Extension with

320\*240 Organic Light Emitting Diode (OLED) 24bit color (Basic OLED graphic library), and Ethernet controller (uIP TCP/IP Stack and WEB server) available at

[http://www.ehitex.de/p\\_info.php?products\\_id=405](http://www.ehitex.de/p_info.php?products_id=405)

[http://www.ehitex.de/p\\_info.php?xPD=113\\_117&products\\_id=405&language=en](http://www.ehitex.de/p_info.php?xPD=113_117&products_id=405&language=en).

**UConnect XE164 OLED and Ethernet Extension Board - Hitex Online-S...**

File Edit View Favorites Tools Help

Back ▾ ▶ Search Folders Favorites

Address [http://www.ehitex.de/p\\_info.php?xPD=113\\_117&products\\_id=405&language=en](http://www.ehitex.de/p_info.php?xPD=113_117&products_id=405&language=en) Go

**hitex**  
DEVELOPMENT TOOLS

You are here: Catalog | USB Application Sticks | Infineon | 70.19.0013.0

**USB Application Sticks**

- ALTERA
- Infineon
- NXP
- Renesas
- ST
- Hilscher

**Starter Kits**

**Application Kits**

**Evaluation Boards**

**CAN**

**I2C**

**Programmer**

**Cortex Debugging**

**HITOP Upgrades**

**MCDS Debugging**

**Software**

**Documentation**

**Accessories**

**All Products**

**Informations**

- Condusion of contract
- Shipping Costs
- Payment methods
- Cancellation
- Redeem Coupon
- About us

**Products [4 at 8]**

**UConnect OLED Extension Board** 149,00 EUR

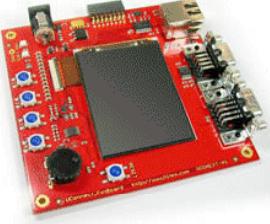
Ready for dispatch in days: 10-15  
 Weight: 0.50 Kg

**Description** **More Images** **Downloads** **Printable**

Extension board for UCONNECT XE164 with OLED display and Ethernet connector.  
 Includes comprehensive application software with source code.

**Technical data:**

- 320\*240 OLED 24bit color (connected to XE164 USIC via SPI)
- Ethernet controller (connected to XE164 USIC via SPI)
- Potentiometer
- 4 Switches
- CAN and UART connector
- Power supply



Click to enlarge

The application in the source code consists of:

- PID regulator
- Basic OLED graphic library
- uIP TCP/IP Stack
- WEB server

Created with HITOP and Tasking classic compiler

**Loginbox**

Login:   
 Password:

[Register](#) [Forgot your password ?](#)

**Search**

ok

**Tell a Friend**

ok

**CALL-BACK**

**Ask a Question**  
**Products Review**  
**Tell a Friend**

1 Stk.

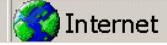
Customers who bought this product also purchased

© Copyright by Hitex | [www.hitex.de](http://www.hitex.de)

Home | Contact | Support | Account | Customers Info | Imprint | Conditions for Sale and Supply

The Layout of the Homepage, graphics and pictures used and the collection of individual contributions are protected by copyright.

All prices are valid only for webshop orders and plus value added tax and shipping costs.



Install the Infineon **DAS** (Device Access Server) Server:

Go to [www.infineon.com/DAS](http://www.infineon.com/DAS):



The screenshot shows a Microsoft Internet Explorer window with the title bar "DAS Tool Interface - Infineon Technologies - Infineon Technologies". The address bar contains the URL "http://www.infineon.com/cms/en/product/promopages/das/index.html". The main content area displays the "DAS Tool Interface" page. On the left, there's a sidebar with the Infineon logo and the word "DAS". The central content includes a "DAS Product Brief" section with download links for "DAS Edition v2.9.2" and "JTAG JDRV LPT Server v2.4.0". Below this is a "Note" section with instructions to use the latest DAS release and report problems to "das.support@infineon.com". To the right, there's a "Download-Service" section listing files for download, each with a link and file size: "das-product-brief.pdf [132 KB]", "DAS Edition v2.9.2.zip [10.57 MB]", "JTAG JDRV LPT Server v2.4.0.zip [4.01 MB]", "SPRINT MCD API V1.0.zip [632 KB]", and "MCD API Press Release.pdf [100 KB]". At the bottom of the page, there are links to "Home", "Company", "Investor", "Press", "Careers", "Infineon worldwide", and copyright information. The status bar at the bottom of the browser window shows the URL "http://www.infineon.com/cms/en/corporate/company/location" and a "Trusted sites" icon.



**Note:**

The DAS Server must be installed on your host computer!

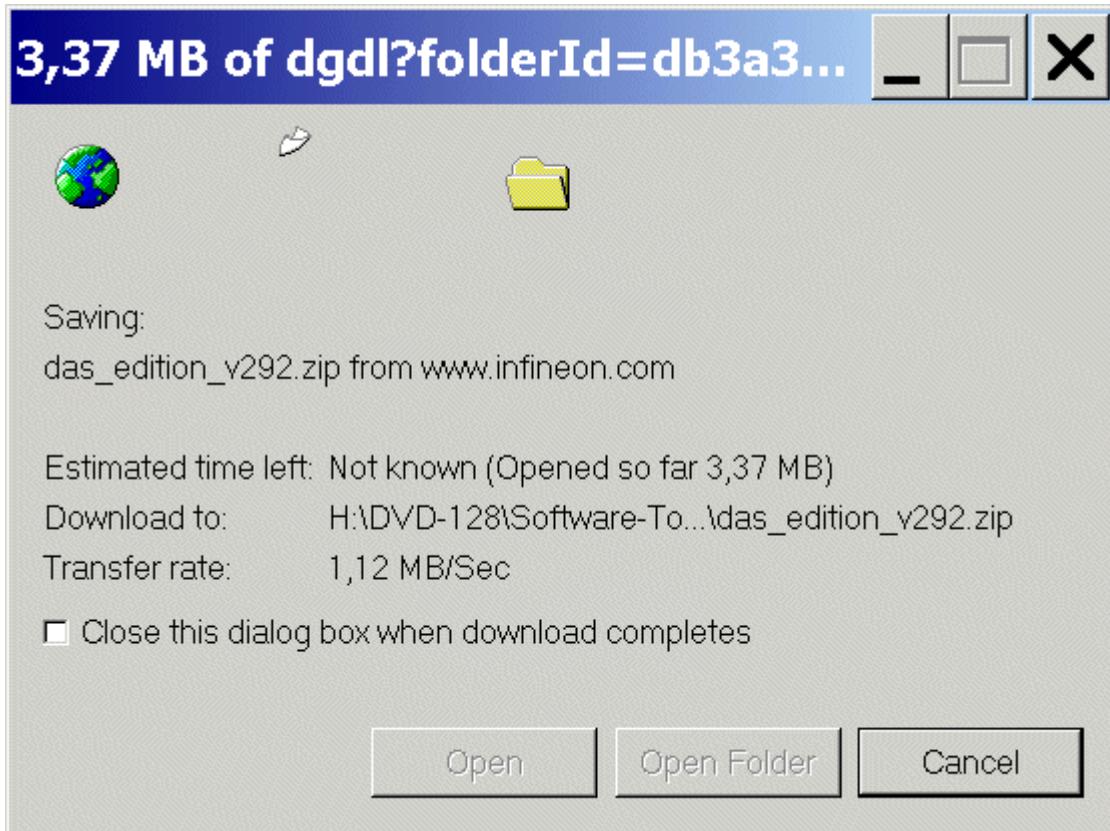
The goal of the DAS software is to provide one single interface for all types of tools (e.g. Programmers/Flash Tools, Emulators/Debugger Development Systems).

The USB Device driver communicates with the UConnect-CAN XE164 when connected to the host computer.

The USB Device driver for the UConnect-CAN XE164 USB interface is included in the DAS software.

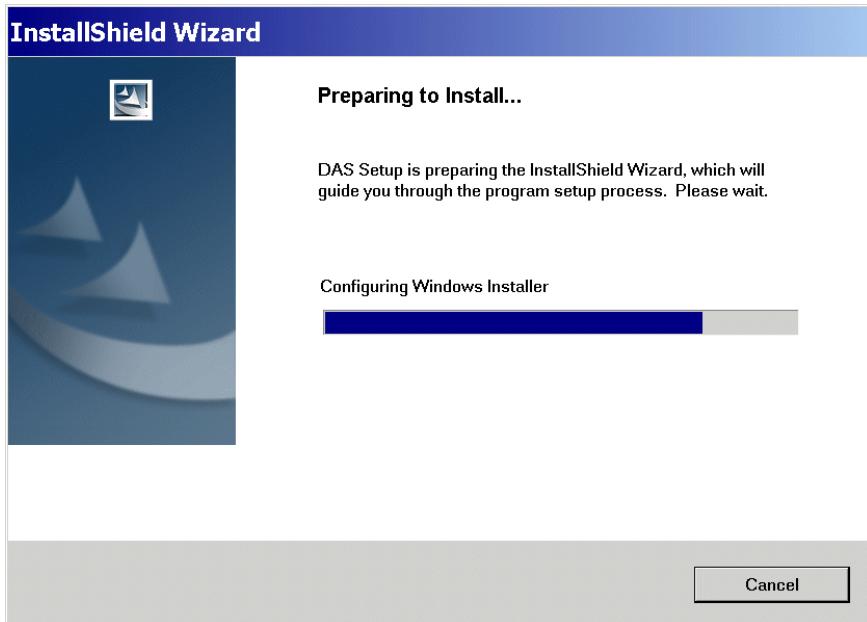
A virtual COM port driver is also included.

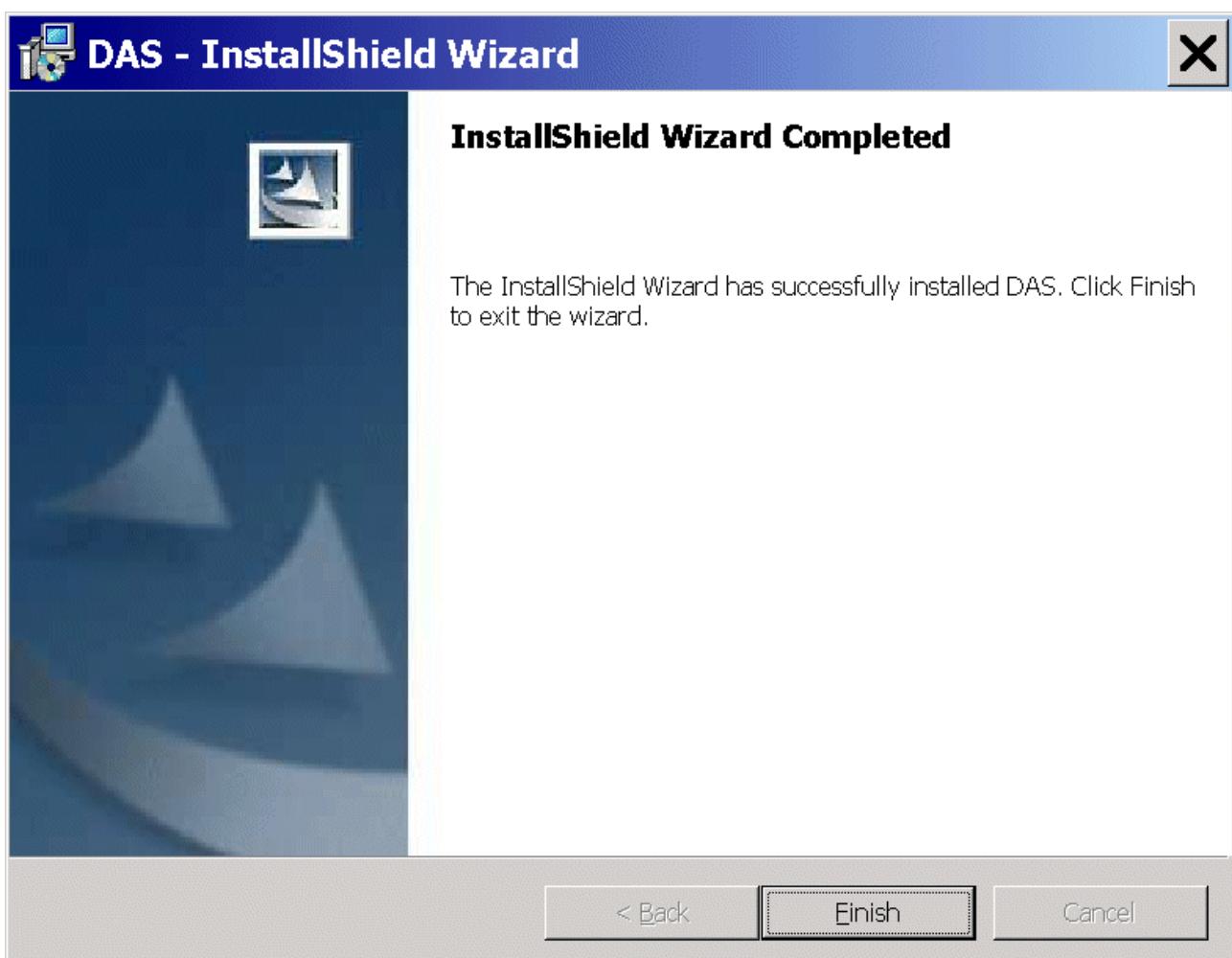
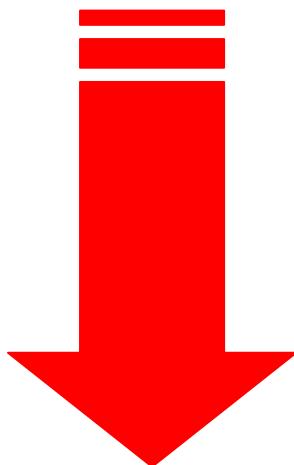
Download “The latest release version for download: DAS Edition v2.9.2”  
( - or any higher version !!! ):



Unzip das\_edition\_v292.zip and

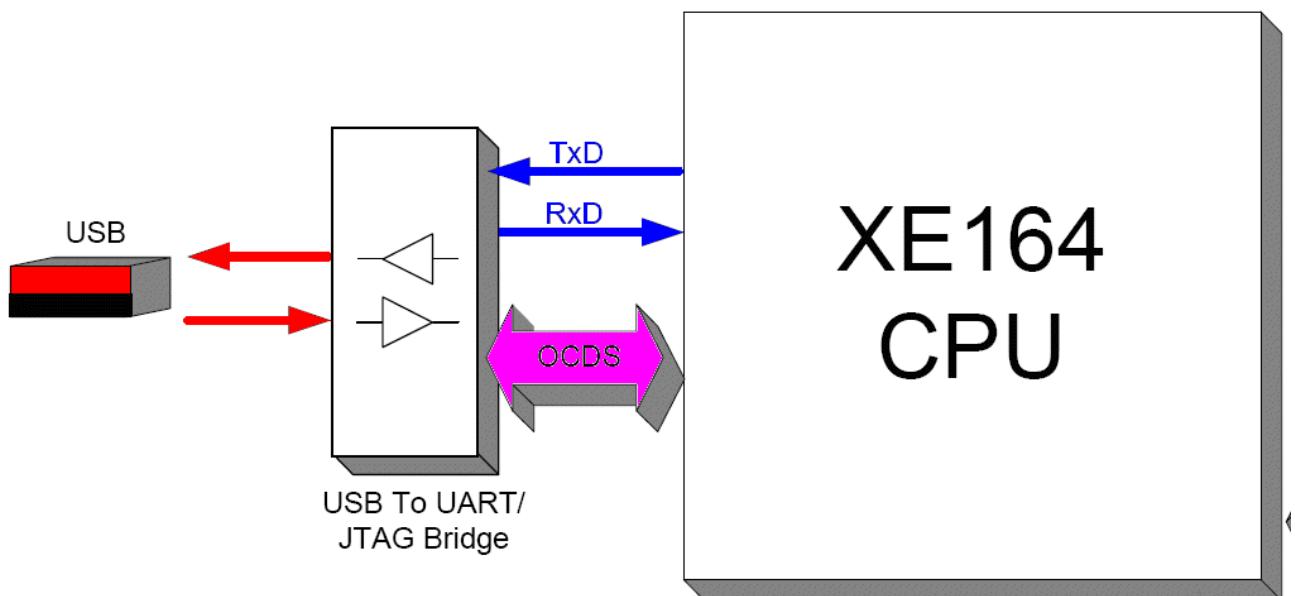
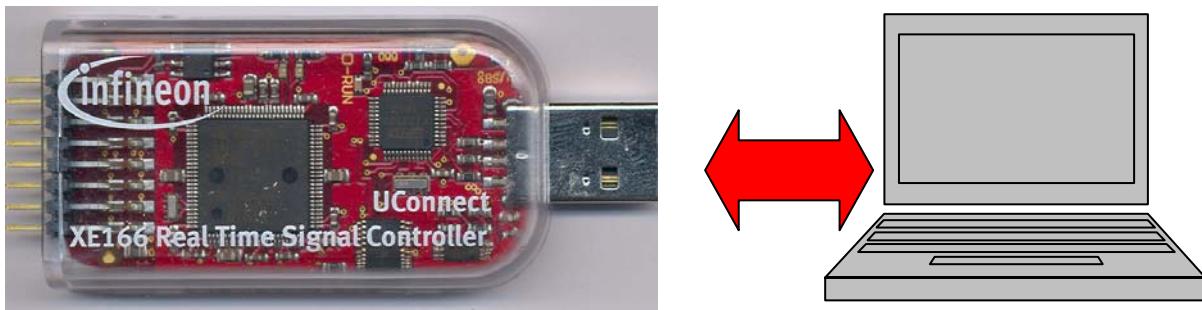
execute “[DAS\\_v292\\_setup.exe](#)” to install the DAS Server.





**Click Finish**

Connect the UConnect-CAN XE164 to the host computer:



### USB Connection:

- .) used for: **UART communication** (the USIC0\_CH0/UART/RS232/serial interface is available via USB as a virtual COM port of the second USB channel of the FTDI FT2232 Dual USB to UART/JTAG interface).
- .) used for: **On-Chip-Flash-Programming and Debugging** (first USB channel of the FTDI FT2232 Dual USB to UART/JTAG interface).
- .) the USB connection works also as the power supply.

## Found New Hardware



DAS JTAG over USB UConnect



**Note:**

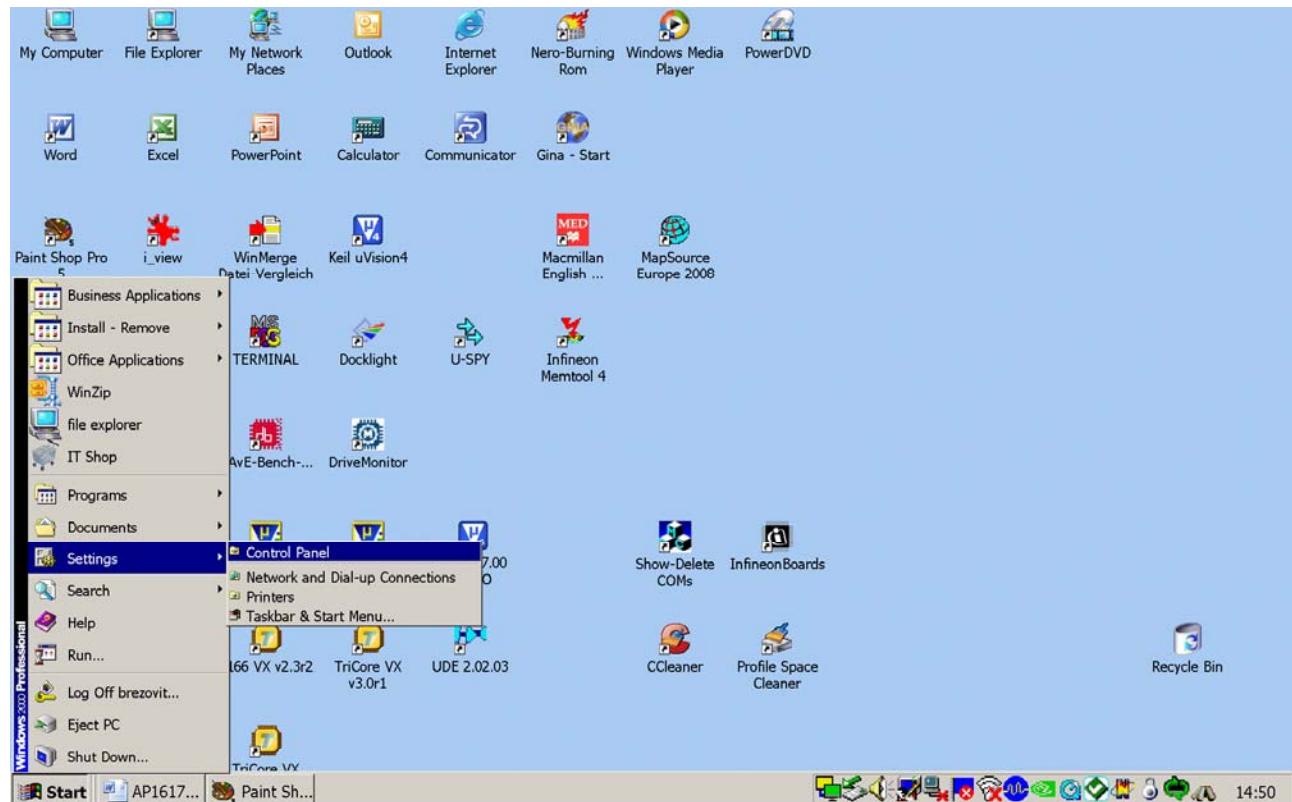
A USB driver is installed the first time while connecting the UConnect-CAN XE164 via USB to your host computer.

**Note:**

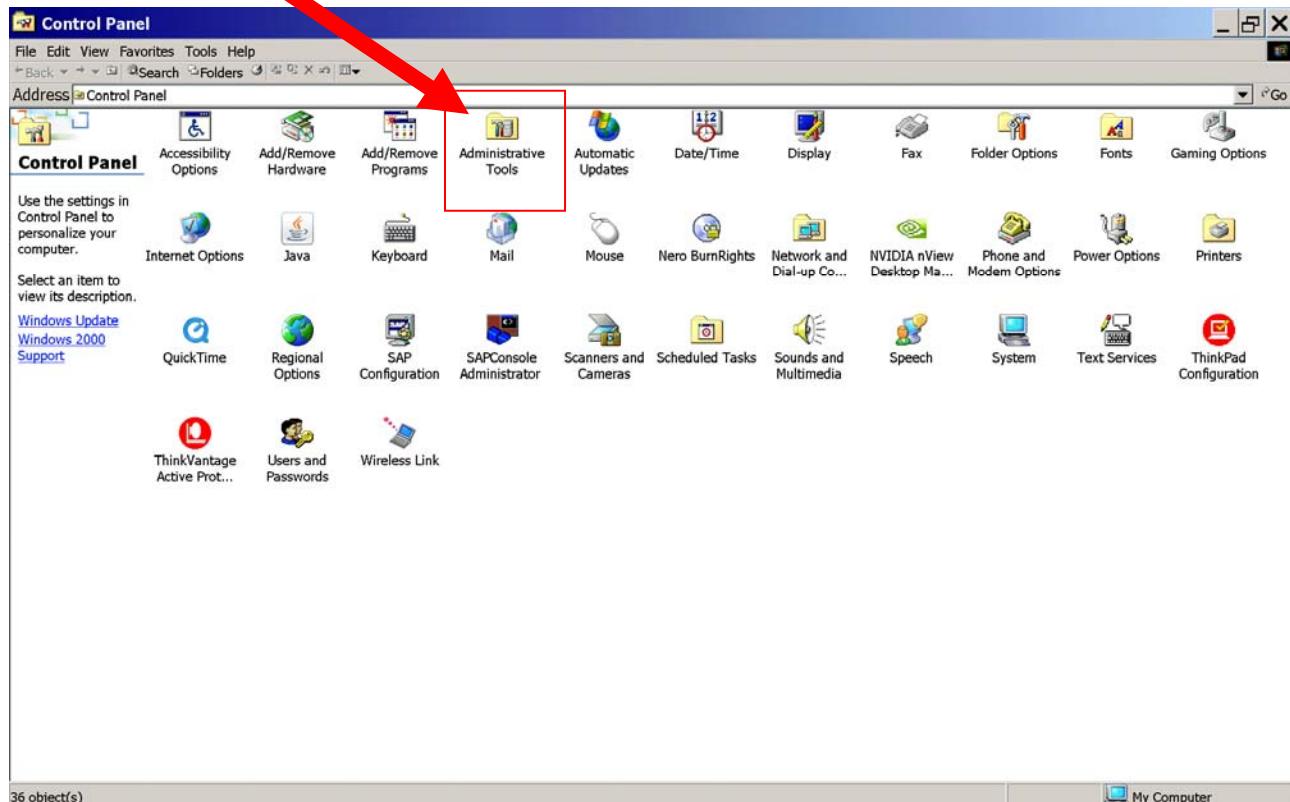
A default virtual COM Port is generated.

Using a Windows 2000 operating system, we are now going to search for the virtual COM Port which was generated after connecting our UConnect-CAN XE164:

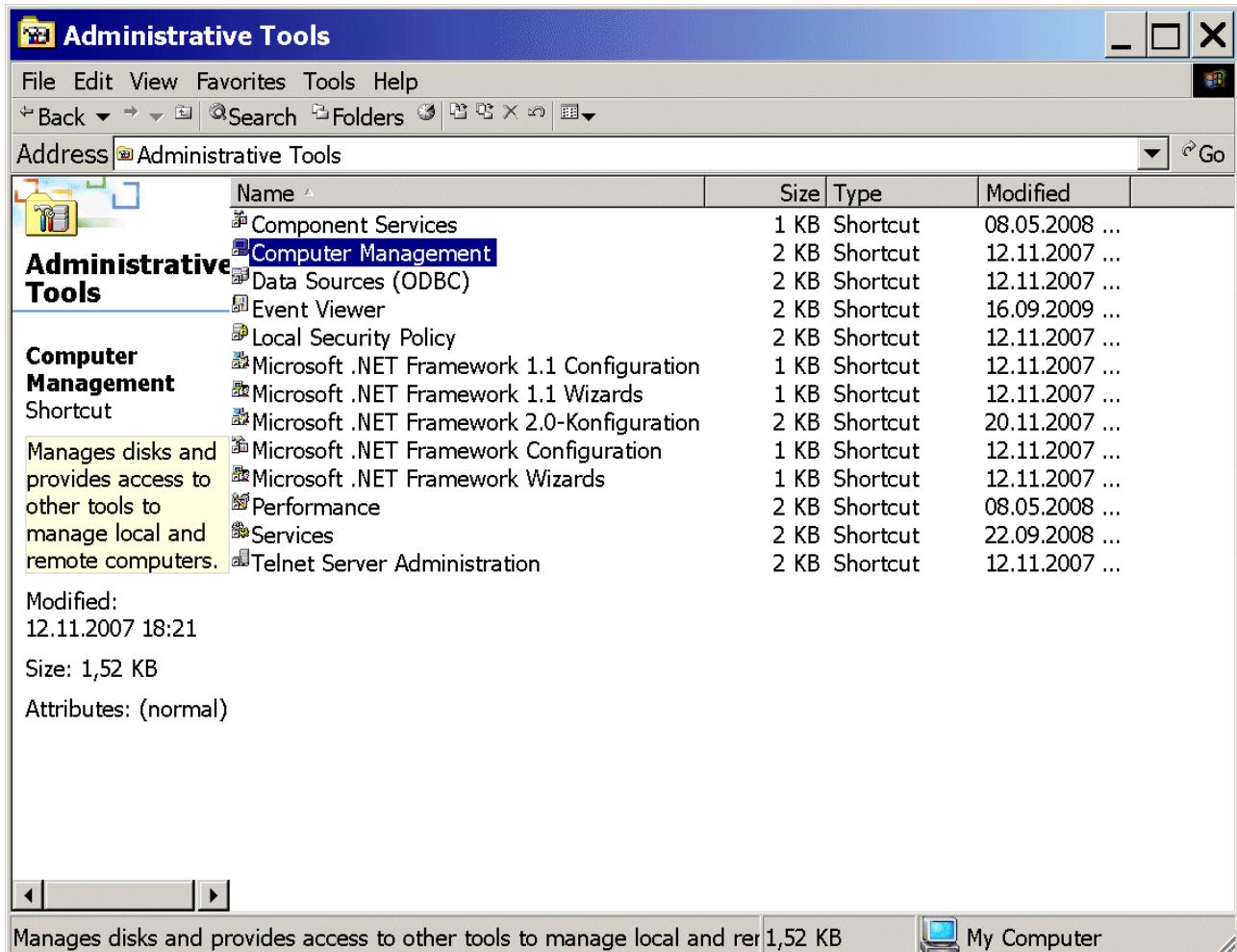
Start – Settings – Control Panel



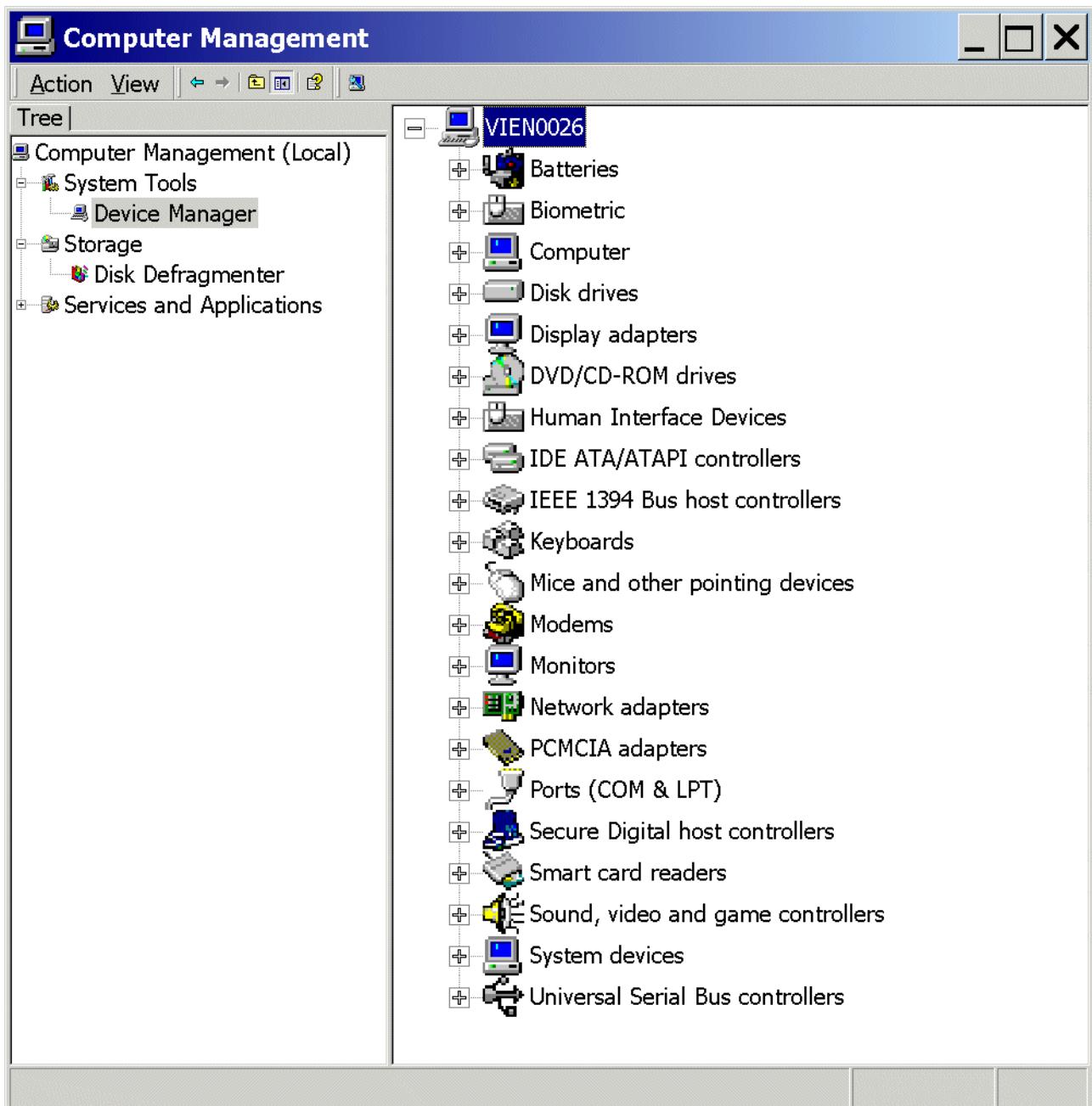
Double click: Administrative Tools



Double click: Computer Management



Click: Device Manager

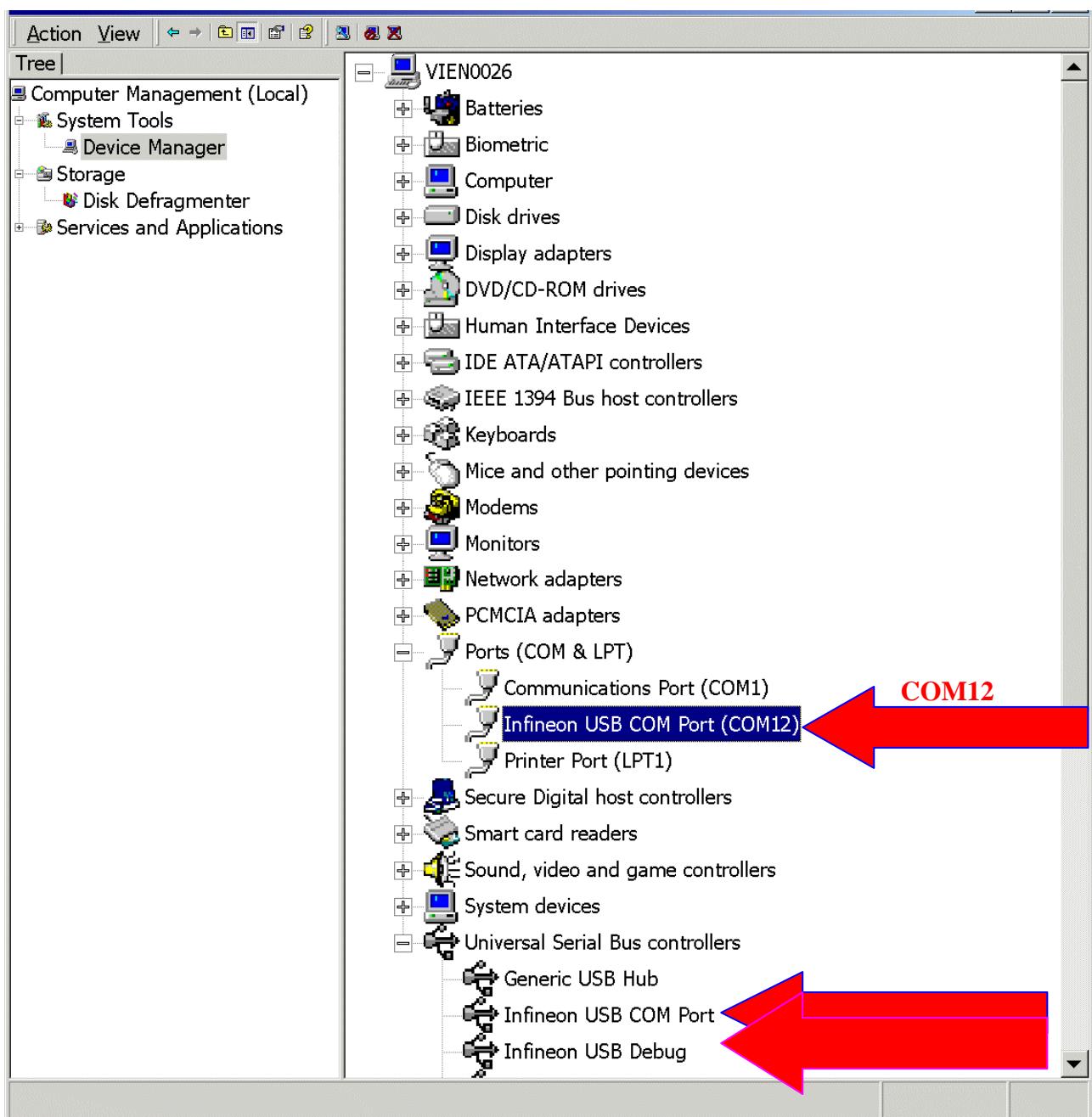


Expand: Ports (COM & LPT):

Expand: Universal Serial Bus controllers:

**Note:**

As we can see:  
our virtual COM Port for  
UART/RS232 communication with the  
UConnect-CAN XE164 via USB is  
**COM12!**

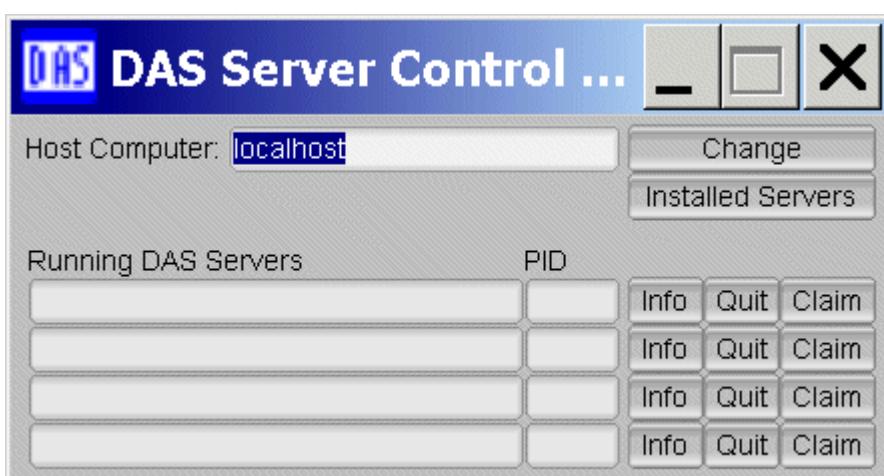




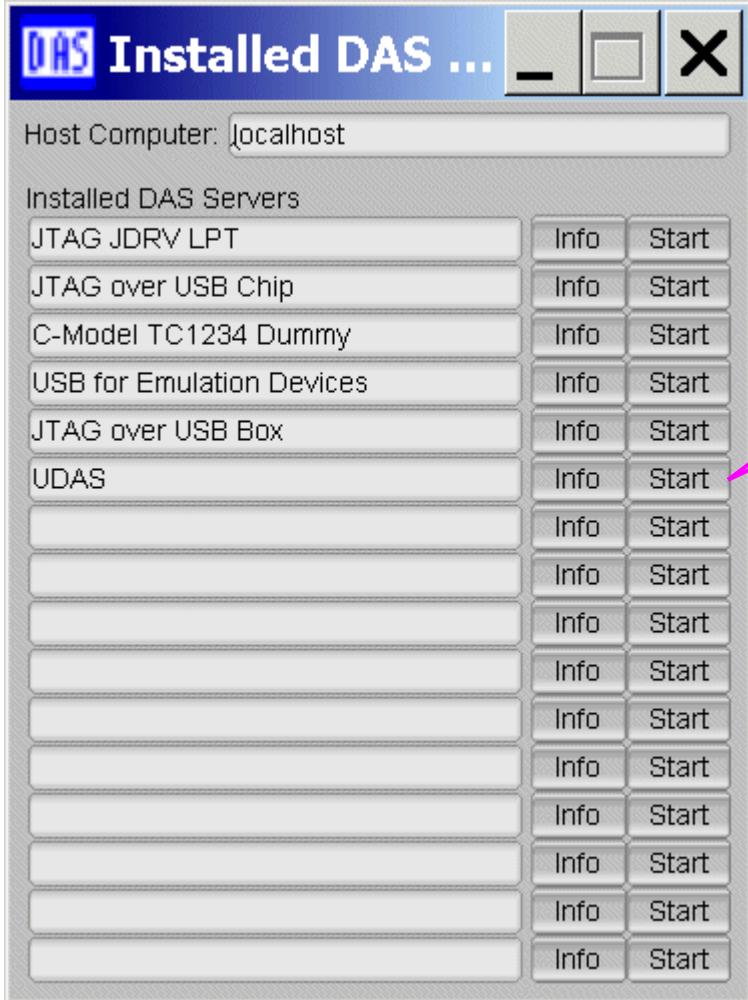
(U)DAS (Universal Device Access Server) Server **hint 1:**

If (U)DAS is not running – even though the board is connected – you may start UDAS with the following dialog:

Start – Programs – DAS – DAS Server Control Panel



**Click** Installed DAS Servers:



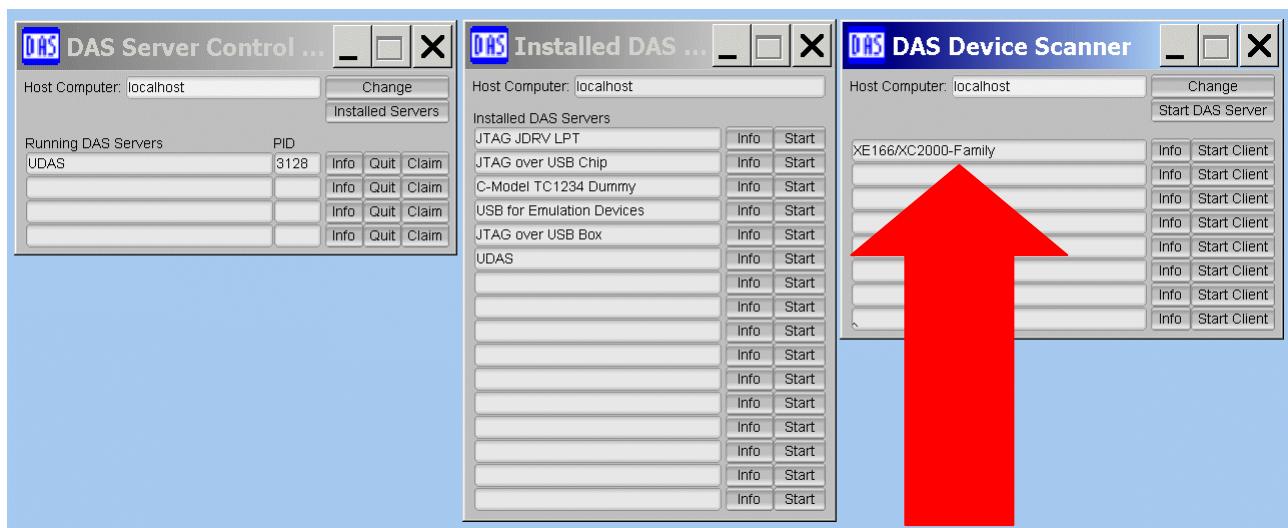
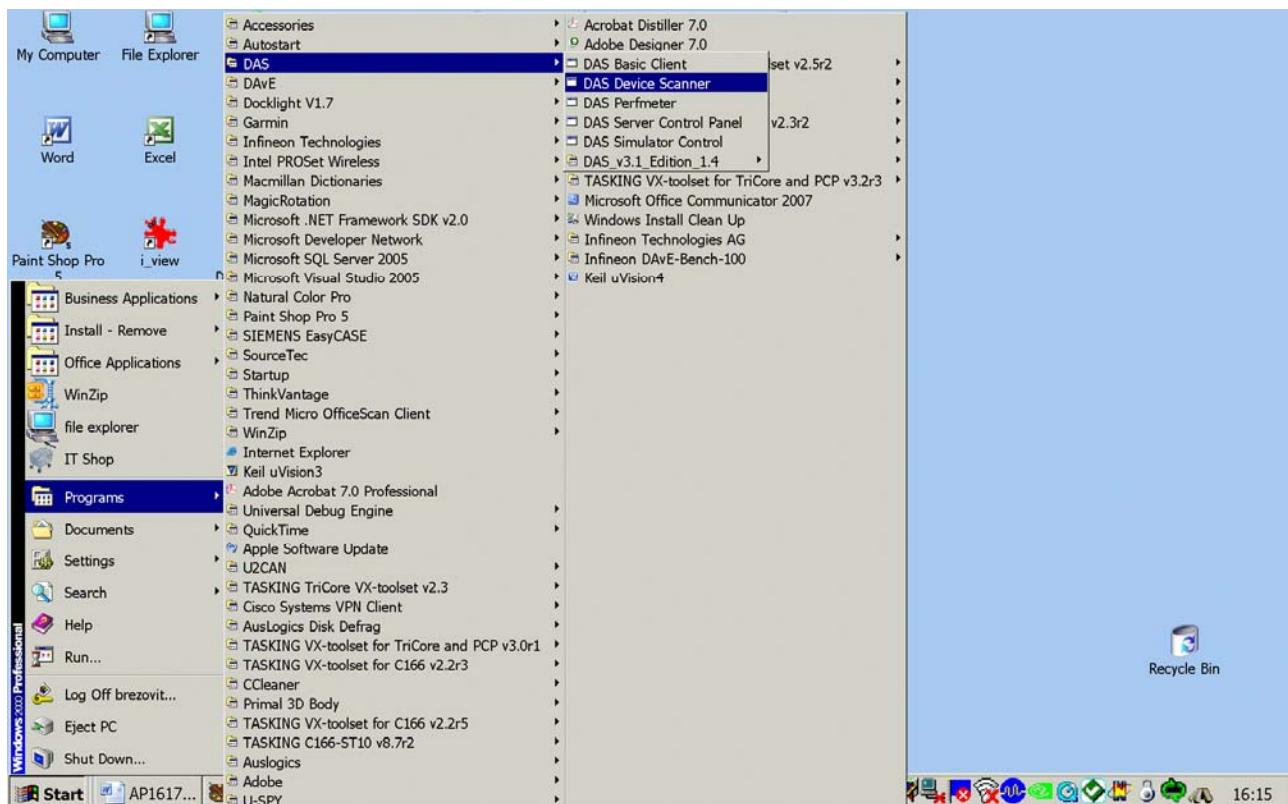
You can  
**click UDAS Start here**



### (U)DAS (Universal Device Access Server) Server hint 2:

With the DAS Device Scanner, you can see if the board has been recognized – to do this - you can use the following dialog:

Start – Programs – DAS – DAS Device Scanner

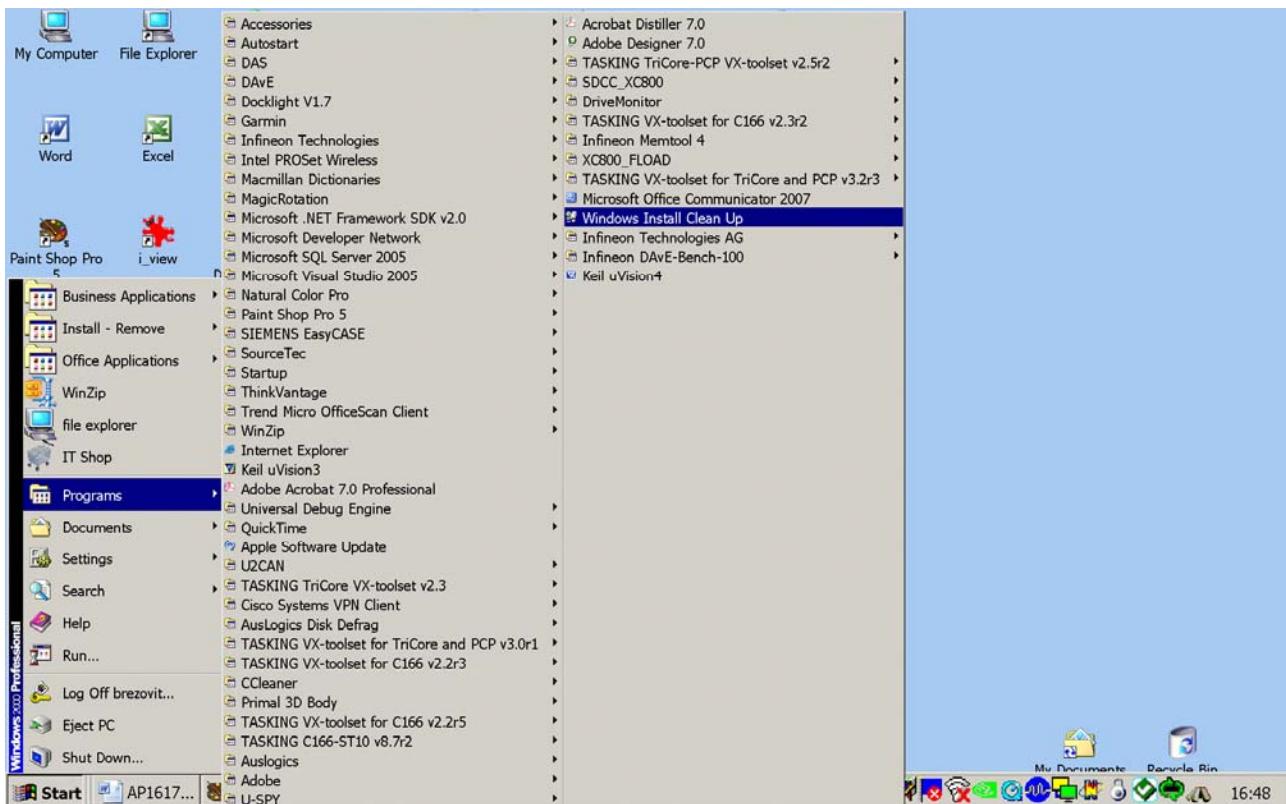


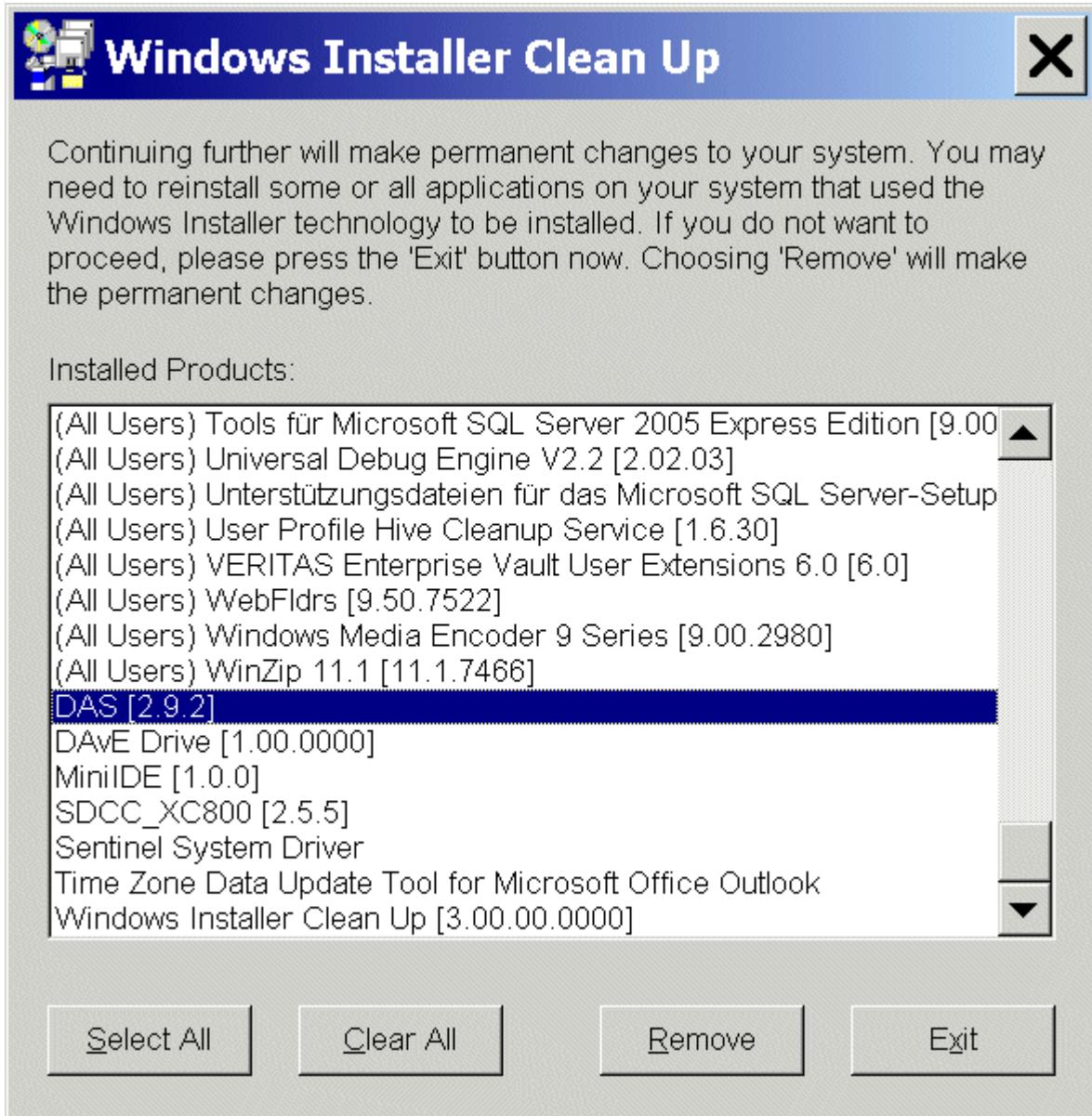


### (U)DAS (Universal Device Access Server) Server hint 3:

If you want to remove DAS you can use the following dialog:

**Execute msicuu2.exe** (to install Windows Install Clean Up)  
**Start – Programs – Windows Install Clean Up**





Select DAS[ ]

Click  Remove

**Note:**

Be careful with this tool!

With only 2 mouse clicks you can remove all software on your computer!



## 2.) DAvE – Installation for XE16x microcontrollers:



### Install DAvE (mothersystem):

Download the DAvE-mothersystem **setup.exe** @ <http://www.infineon.com/DAvE>

Title	Date	Version	Size
<b>Tool Package</b>			
 DAvE - Mothersystem (DAvE_Mothersystem_v2_2r1.zip)	14 Dec 2009	V2.2	8.8 MB
 DAvE - Mothersystem (setup.exe)	14 Dec 2009	V2.2	8.9 MB

and execute **setup.exe** to install DAvE .

### Note:

Abort the installation of Acrobat Reader.



**Install the XE164 microcontroller support/update (XE16xx\_Series.dip):**

1.)

Download the DAvE-update-file (**.DIP**) for the required microcontroller  
 @ <http://www.infineon.com/DAvE>

**DAvE - Infineon Technologies - Infineon Technologies**

File Edit View Favorites Tools Help

Back Search Folders Favorites Home | Where to Buy | Sitemap | Select Language | Login About Infineon >

Address: http://www.infineon.com/cms/en/product/channel.html?channel=db3a3043156fd57301161fddead71f5b Go

infineon

Get Product information Select a Category Microcontrollers

Home > Microcontrollers > Development Tools, Software and Kits > XE166 Development Tools and Software > DAvE

DAvE

DAvE for the Infineon XE166 microcontroller Family

DAvE supports the 16-bit derivatives as DAvE Integration Package (DIP) files.  
 - All the latest DIP's are available for FREE download.

Company Name and Weblink	Product Name	XE167 Series	XE164 Series	Description
	DAvE	x	x	DAvE stands for Digital Application Virtual Engineer and is Infineon Technologies' code generator for their range of 8, 16 and 32 Bit Microcontrollers. It provides initialization, configuration and driver code to ease programming for beginners as well as experts.

Ask Infineon!  
 International Toll Free:  
 0(0)800 951 951 951  
 Direct Access:  
 +49 89 234 65555  
 Infineon is happy to help you:  
 ► Infineon Service Center

Print Page Send Page

DAvE home Documents Contact us

Document Types  
 ▾ Development Tools

Title	Date	Version	Size
XE16xx-Series DIP file for DAvE (Microcontroller Configuration Tool)-latest version (DAvE_XE16xx_Series_v2_1.zip)	24 Aug 2009	v2.1	4.1 MB
XE16xxM-Series DIP file for DAvE (Microcontroller Configuration Tool) (DAvE_XE16xxM_Series_v2_0_Release.zip)	23 Jul 2009	v2.0	4.8 MB
XE16xxN-Series DIP file for DAvE (Microcontroller Configuration Tool) (DAvE_XE16xxN_Series_v0_2.zip)	04 Feb 2010	v0.2	4.2 MB

DAvE\_XE16xx\_Series\_v2\_1.zip

Home | Company | Investor | Press | Careers | Infineon worldwide  
 © 1999 - 2010 Infineon Technologies AG - Usage of this website is subject to our Usage Terms - Imprint - Contact - Privacy Policy

Trusted sites

Unzip the zip-file “**DAvE\_XE16xx\_Series\_v2\_1.zip**” and save “**XE16xx\_Series\_v2.1.dip**“  
 @ e.g. C:\DAvE\XE16xx\_Series\_v2.1.dip.

2.)

Start DAvE - ( click **DAvE** )

3.)

**View****Setup Wizard**

Default: • Installation

Forward&gt;

Select: • I want to install products from the DAvE's web site

Forward&gt;

Select: C:\DAvE

Forward&gt;

Select: Available Products

click ✓ XE16xx\_Series

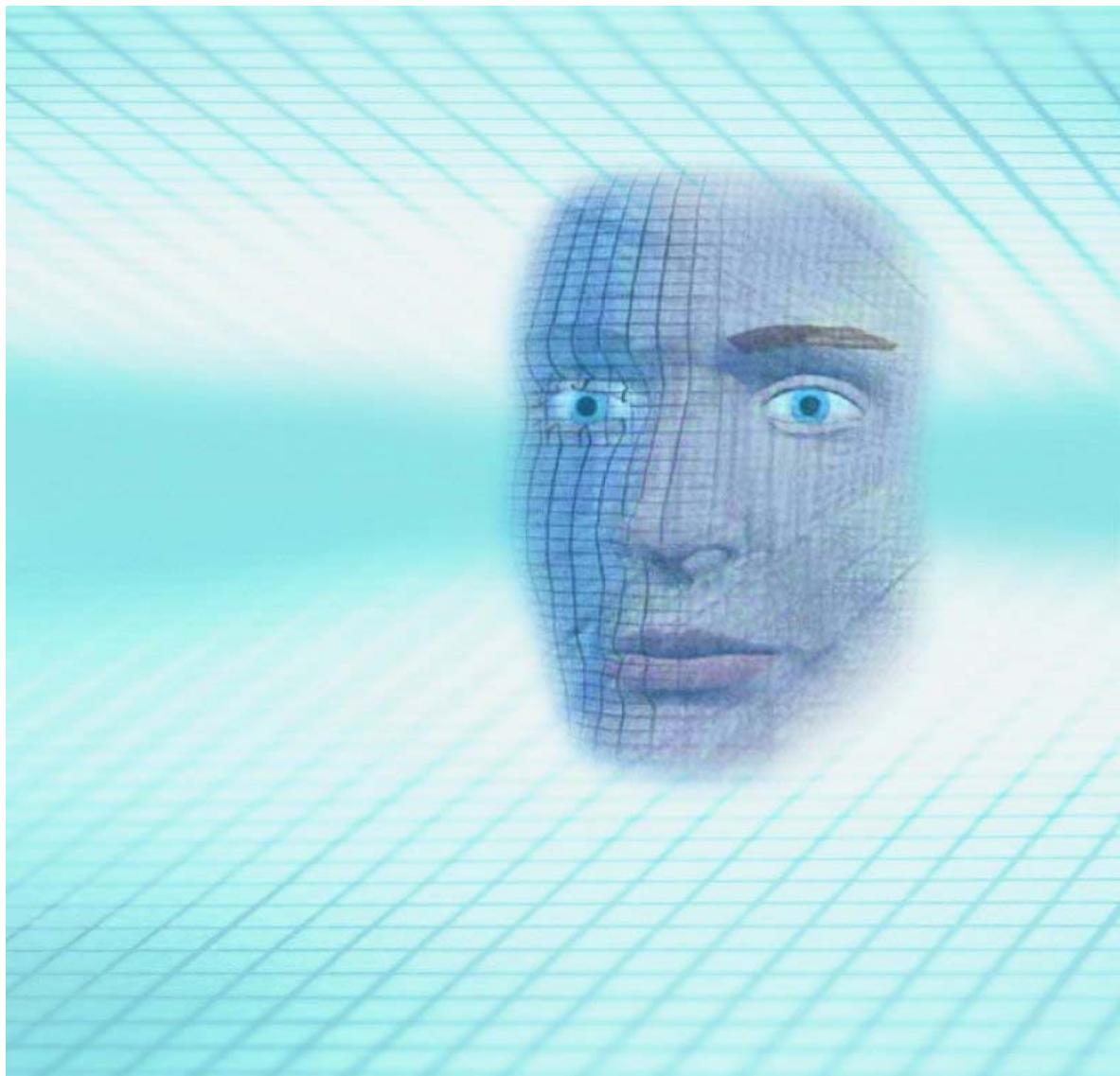
Forward&gt;

Install

End

4.) DAvE is now ready to generate code for the XE16x microcontrollers.

**3.) DAvE - Microcontroller Initialization after Power-On:**





Start the program generator DAvE and select the XE164 microcontroller:

**View - Project Window** (Closes the Project Window)

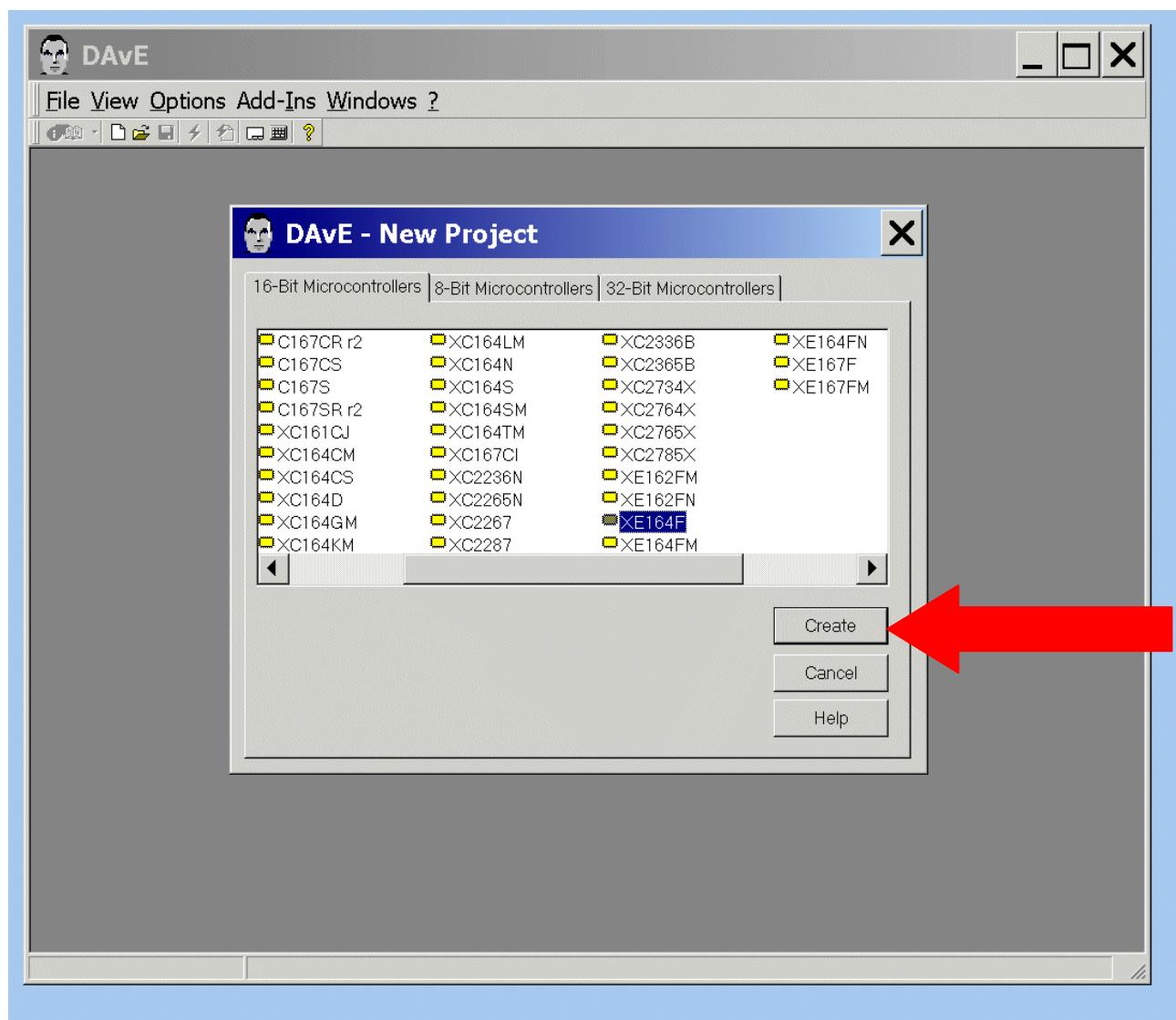
**View - Command Window** (Closes the Command Window)

**File - New**

DAvE – New Project: **click** 16-Bit Microcontrollers

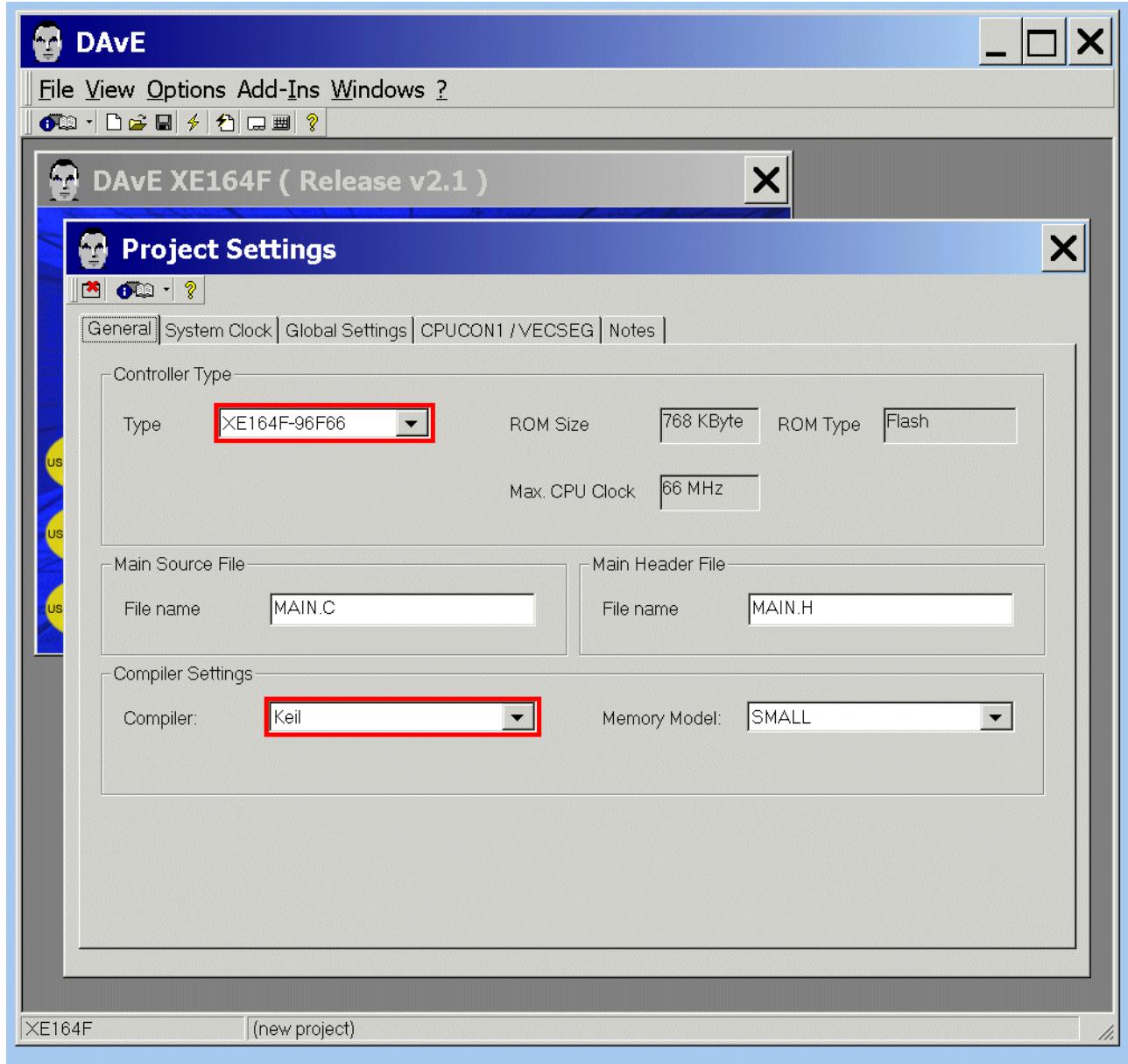
Select **XE164F**

**Create**



Choose the Project Settings as you can see in the following screenshots:

Project Settings: General: Controller Type: Type: **check/select** XE164F-96F66  
Project Settings: General: Compiler Settings: Compiler: **check/choose** Keil



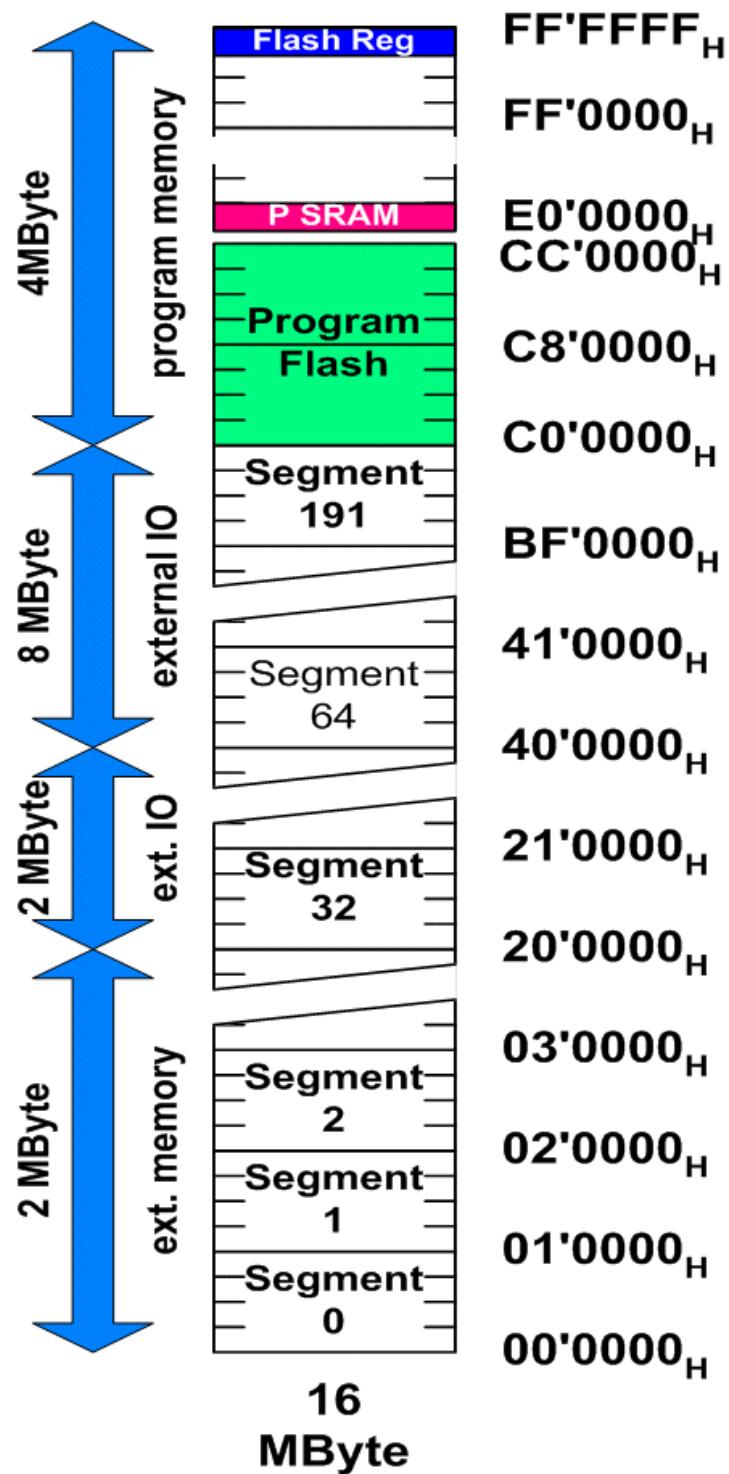
**Note:**

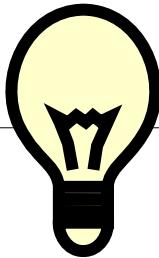
You can change file names (e.g. MAIN.C, MAIN.H) anytime.





Additional information: [Address Space](#) and Memory Model:





Additional information: Address Space and **Memory Model**:

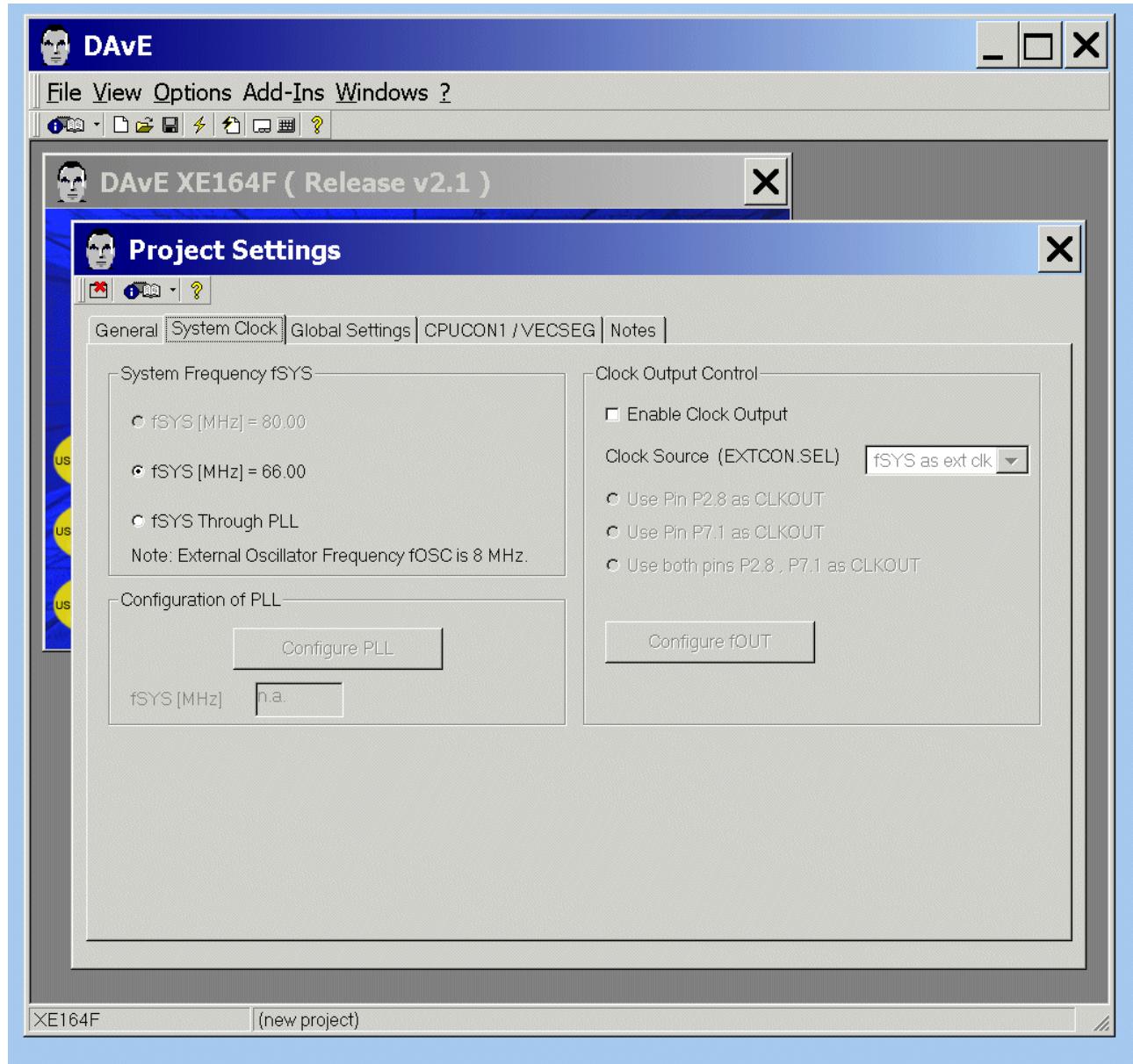
#### Memory models:

TASKING memory model				Default function memory qualifier <sup>1</sup>				
Keil	Data space	Data object	Code space	TASKING		Data space	Data object	Code space
TINY	<64K	<16K	<64K	⇒	TINY	<64K	<64K	<64K
SMALL <sup>2</sup>	<64K	<16K	<64K	⇒	SMALL <sup>2</sup> I,II or III	<64K	<64K	<16M
MEDIUM <sup>2</sup>	<64K	<16K	<16M	⇒	SMALL <sup>2</sup> I,II or III	<64K	<64K	<16M
COMPACT	<16M	<16K	<64K	⇒	MEDIUM	<16M	<16K	<64K
LARGE	<16M	<16K	<16M	⇒	LARGE <sup>3</sup>	<16M	<16K	<16M
HCOMPACT	<16M	<64K	<64K	⇒	MEDIUM	<16M	<16K	<64K
HLARGE	<16M	<64K	<16M	⇒	LARGE <sup>3</sup>	<16M	<16K	<16M
XLARGE	<16M	<16M	<16M	⇒	HUGE	<16M	<16M	<16M

#### Memory qualifiers for variables:

Keil	TASKING <sup>1</sup>	Description
bdata	_bita	bit-addressable RAM
ebdata	_bita	extended bit-addressable memory
idata	_iram	internal RAM
sdata	_system	system page
near	_near	Keil: always restricted to 16K TASKING: 16K or 64K, depends on memory
far	_far	one object is limited to 16K
huge	_shuge	one object is limited to 64K
xhuge	_huge	objects are limited to available memory space

Project Settings: System Clock: (do nothing)

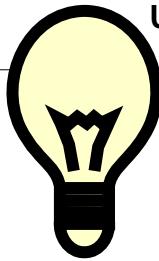


#### Note (Source: DAvE):

Configuration of the System Clock:

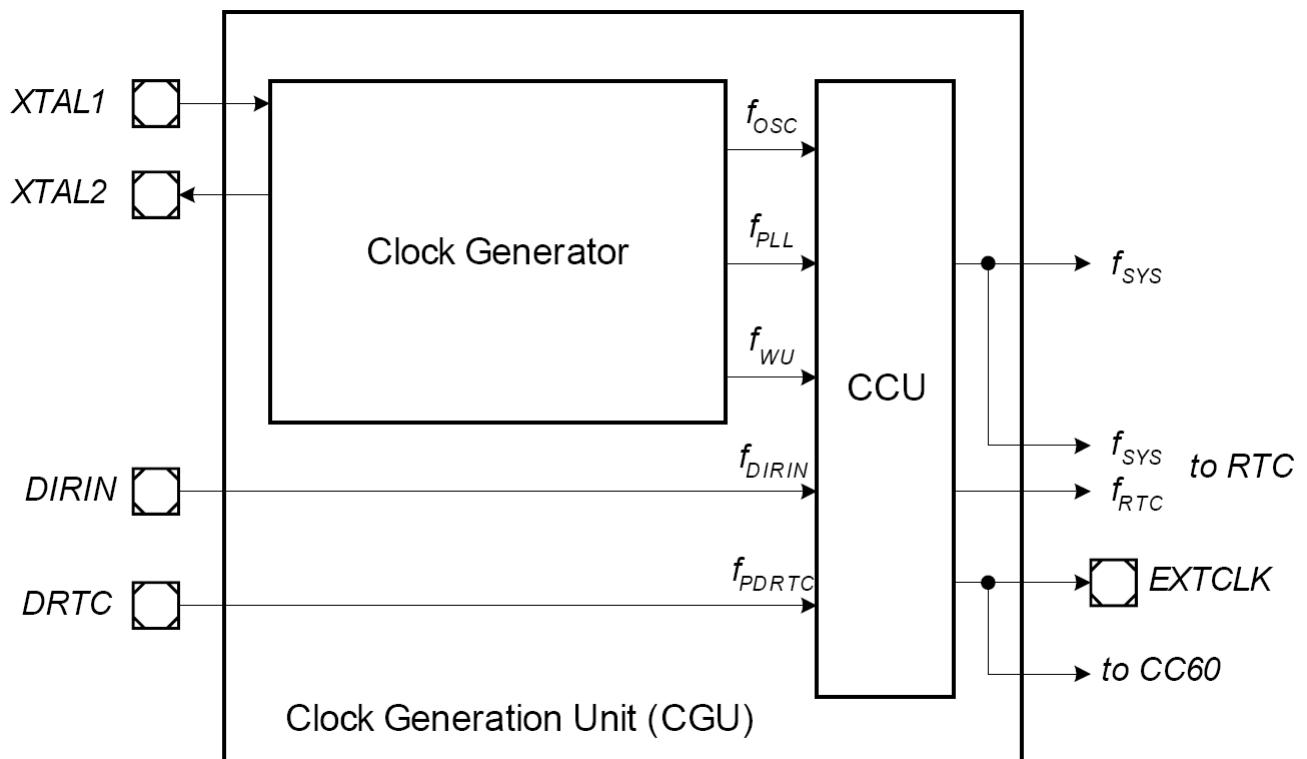
- VCO clock used, input clock is connected
- input frequency is 8,00 MHz (XTAL1)
- configured system frequency is 66,00 MHz
- system clock is 66.00 MHz





Additional information: **Clock System** (Source: User's Manual):

Clock Generation Unit (CGU) Block Diagram:



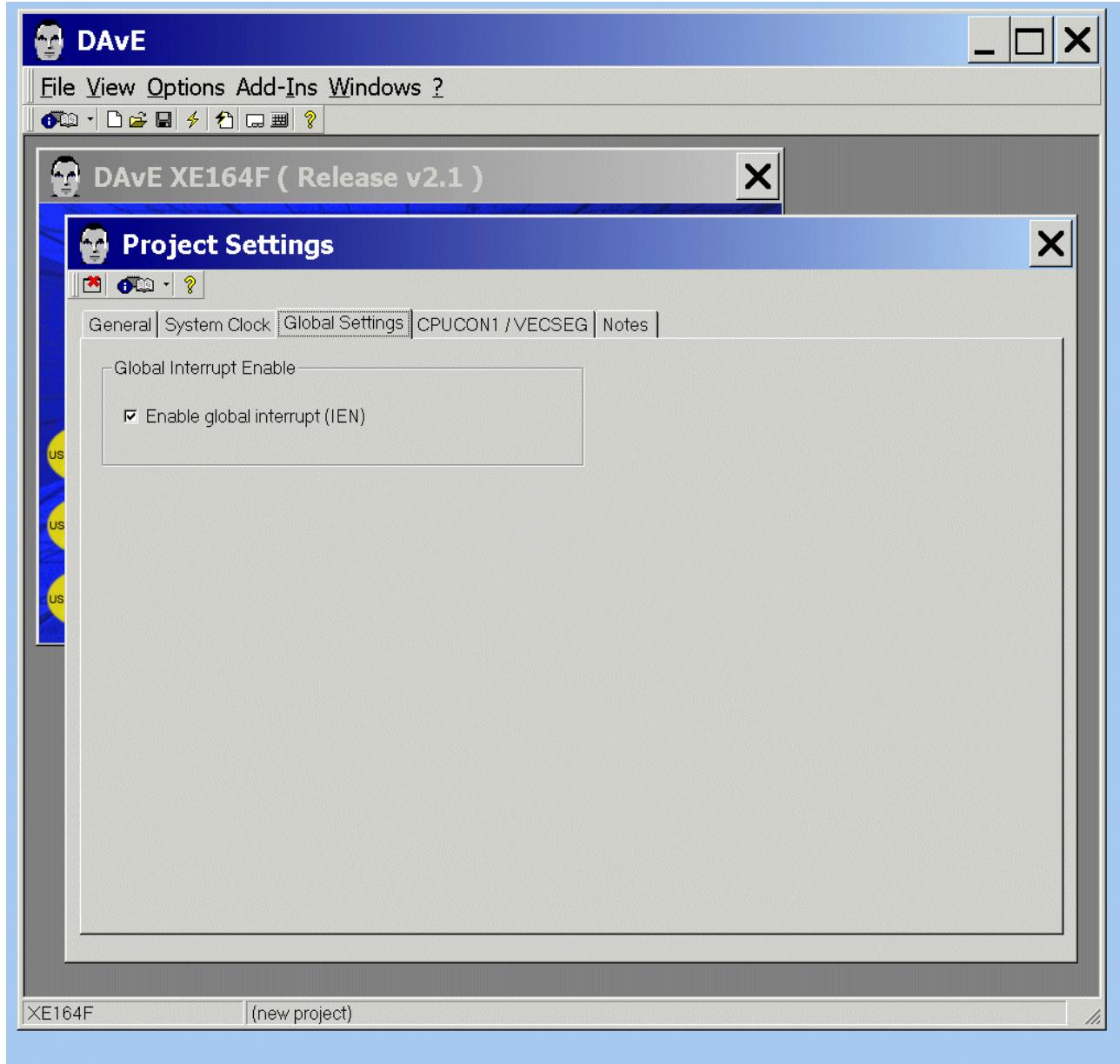
**Note:**

The CGU can convert a low-frequency external clock to a high-speed internal clock, or can create a high-speed internal clock without external input.

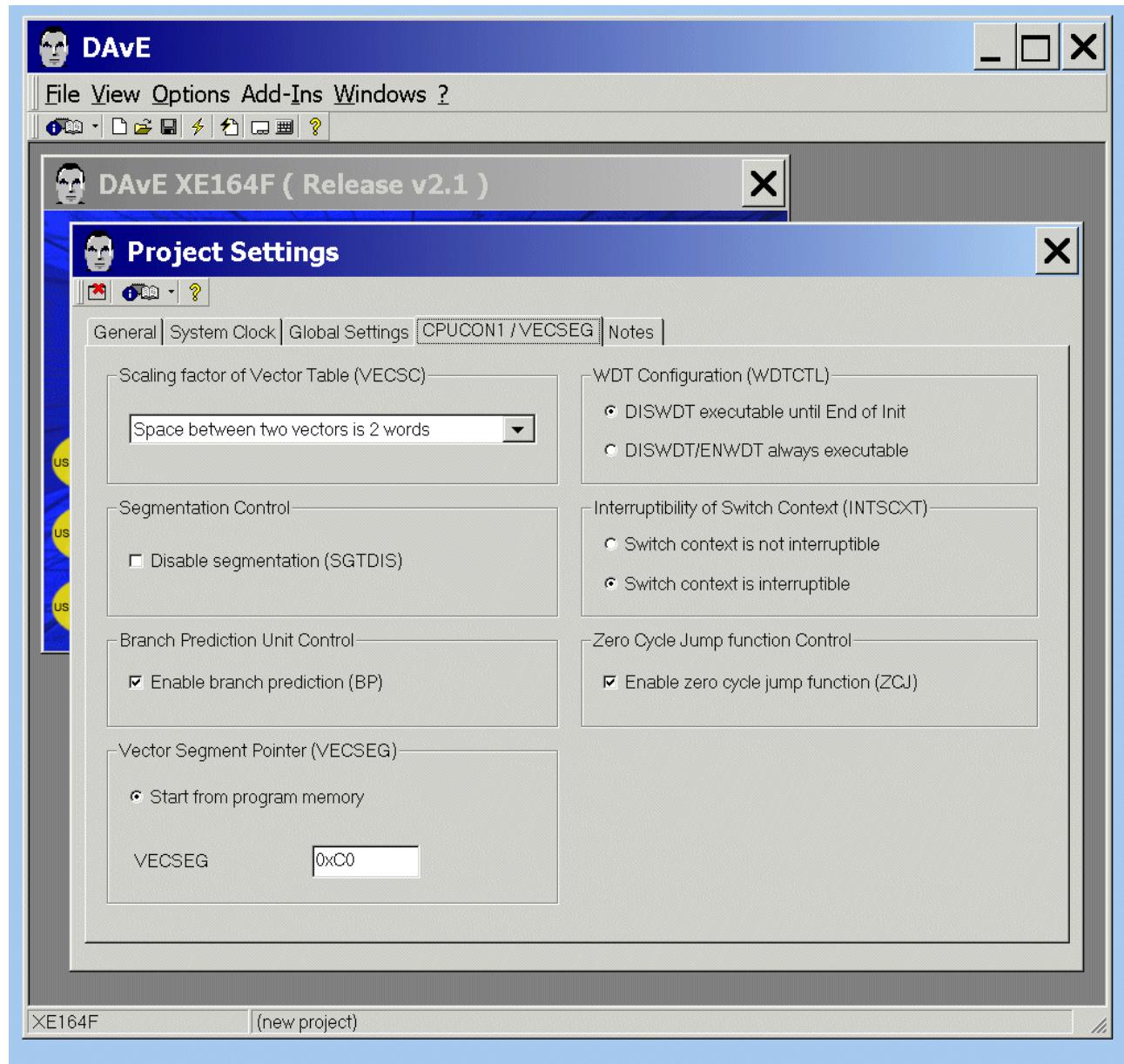
The system clock  $f_{SYS}$  is generated out of four selectable clocks:

- PLL clock  $f_{PLL}$
- Wake-Up clock  $f_{WU}$
- The Direct Clock  $f_{OSC}$ , from pin XTAL1
- Input DIRIN as Direct Clock Input  $f_{DIR}$

Project Settings: **Global Settings:** (do nothing. Do not change configuration)

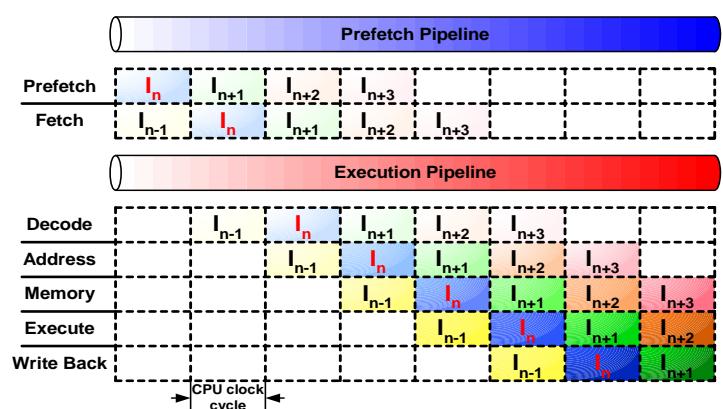


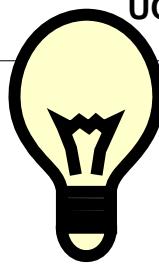
Project Settings: CPUCON1/VECSEG: (do nothing)



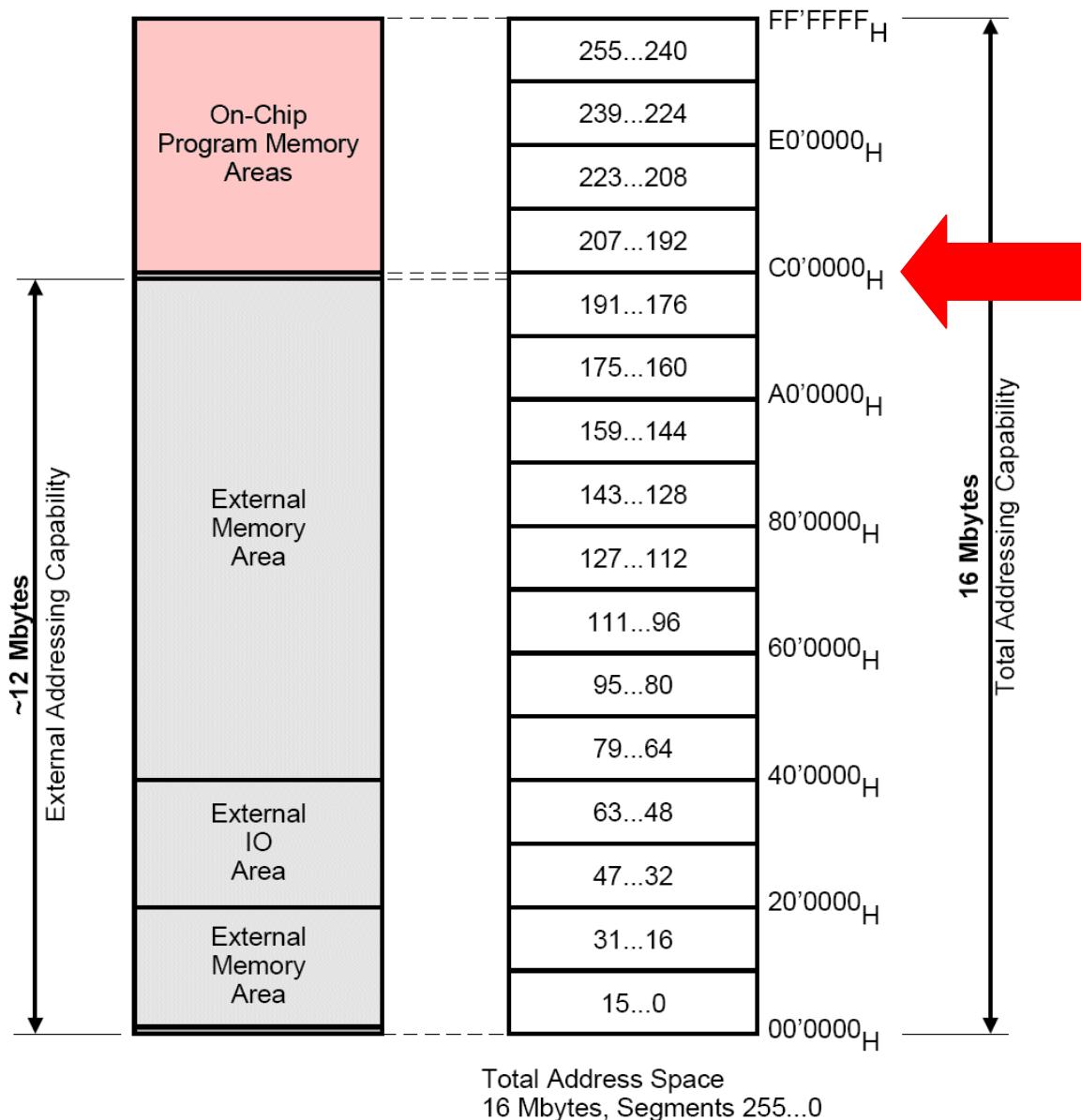
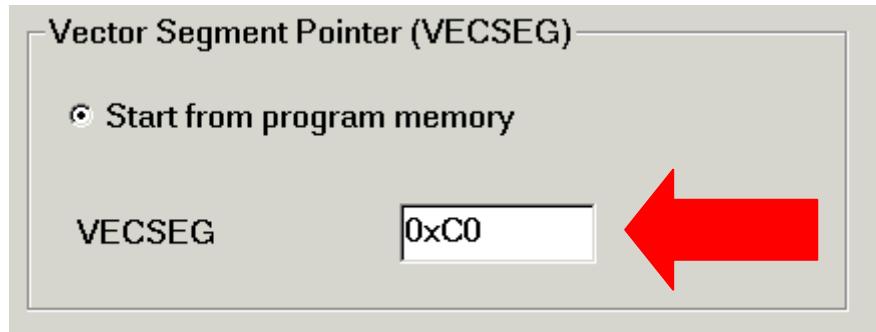
**Note:**

We should not change the pipeline behaviour.

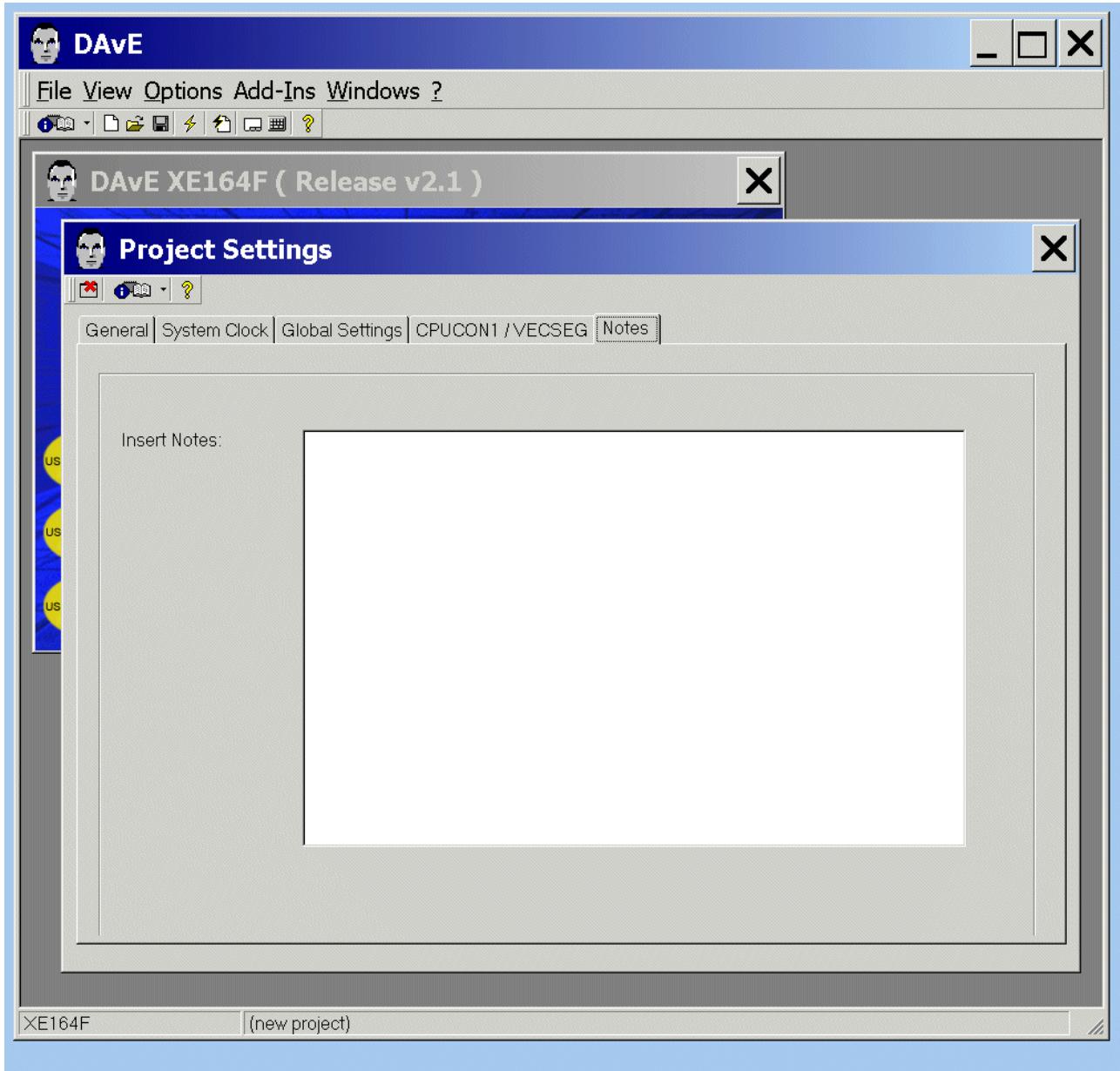




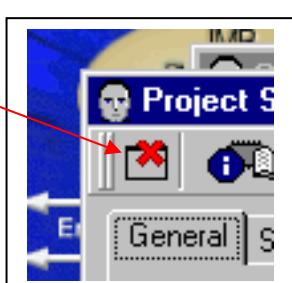
Additional information: **Start from program memory** (Source: User's Manual):



Project Settings: Notes: Insert Notes: If you wish, you can insert your comments here.

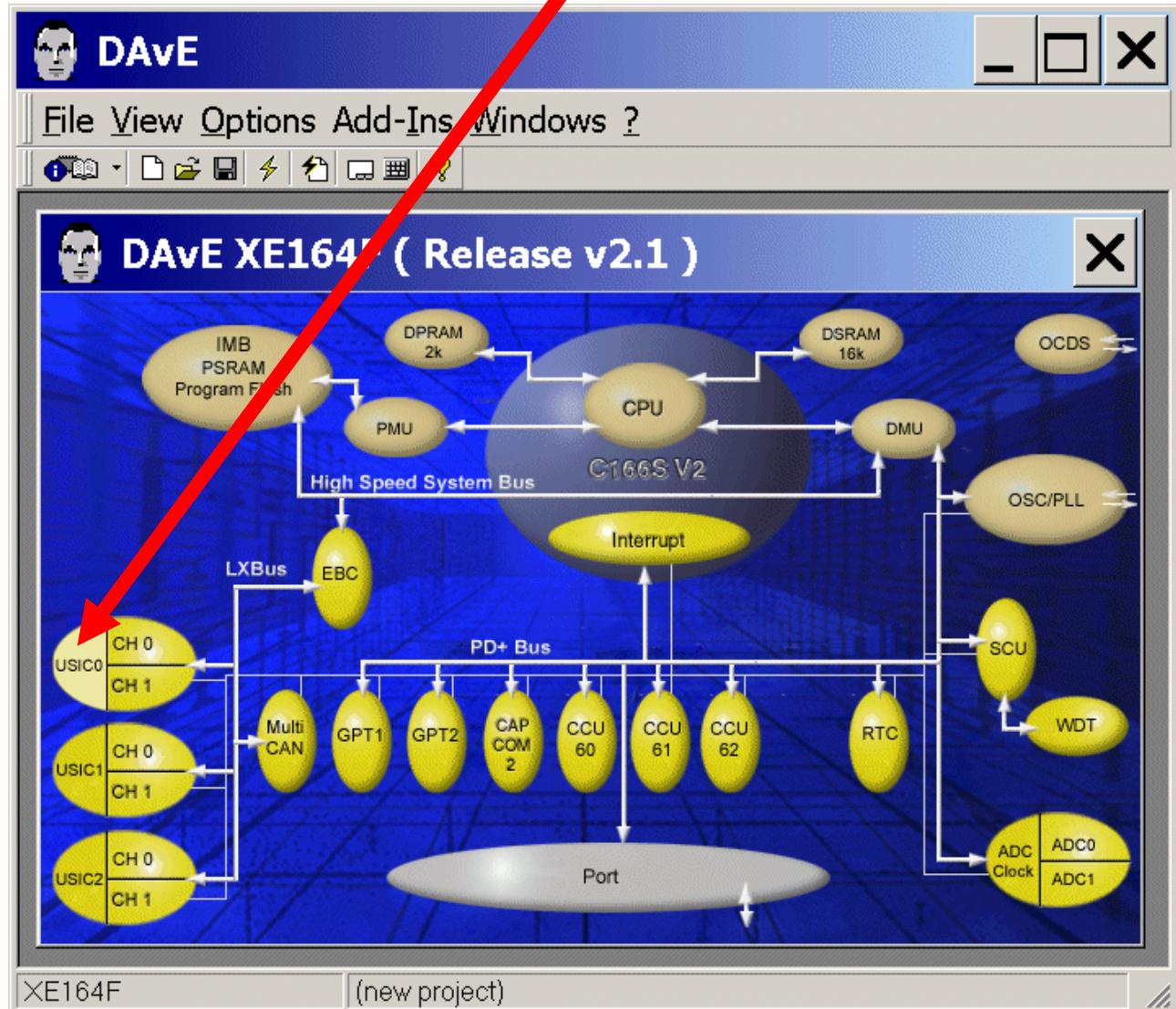


Exit and Save this dialog now by clicking  the close button:



Configuration of the serial interface (Universal Serial Interface Channel module)  
"ASC0" / UART / USIC0 CH0 / U0C0:

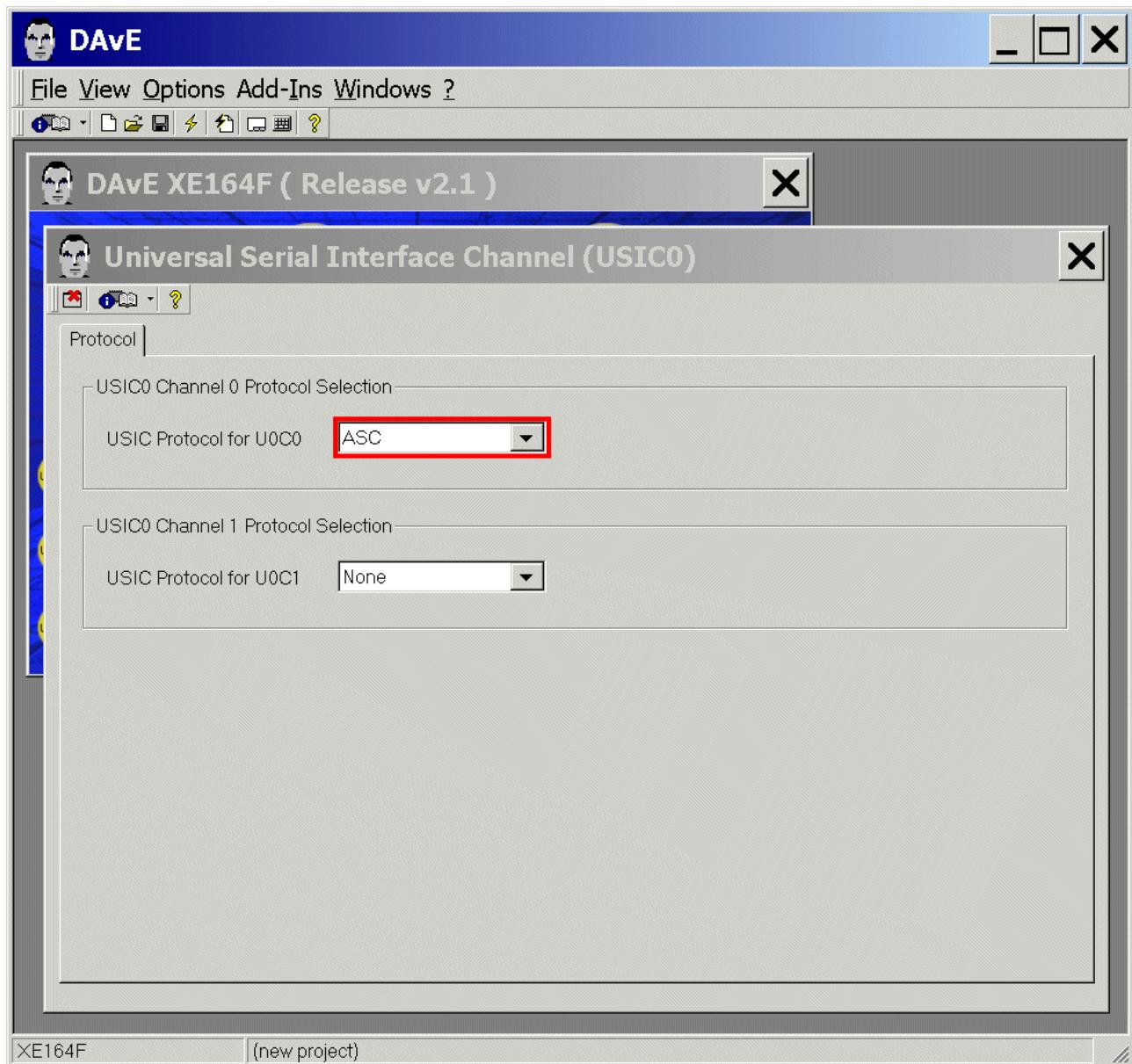
The configuration window/dialog can be opened by clicking the specific block/module (USIC0).



**Note:**

For serial communication with a terminal program running on your host computer U0C0 is used.

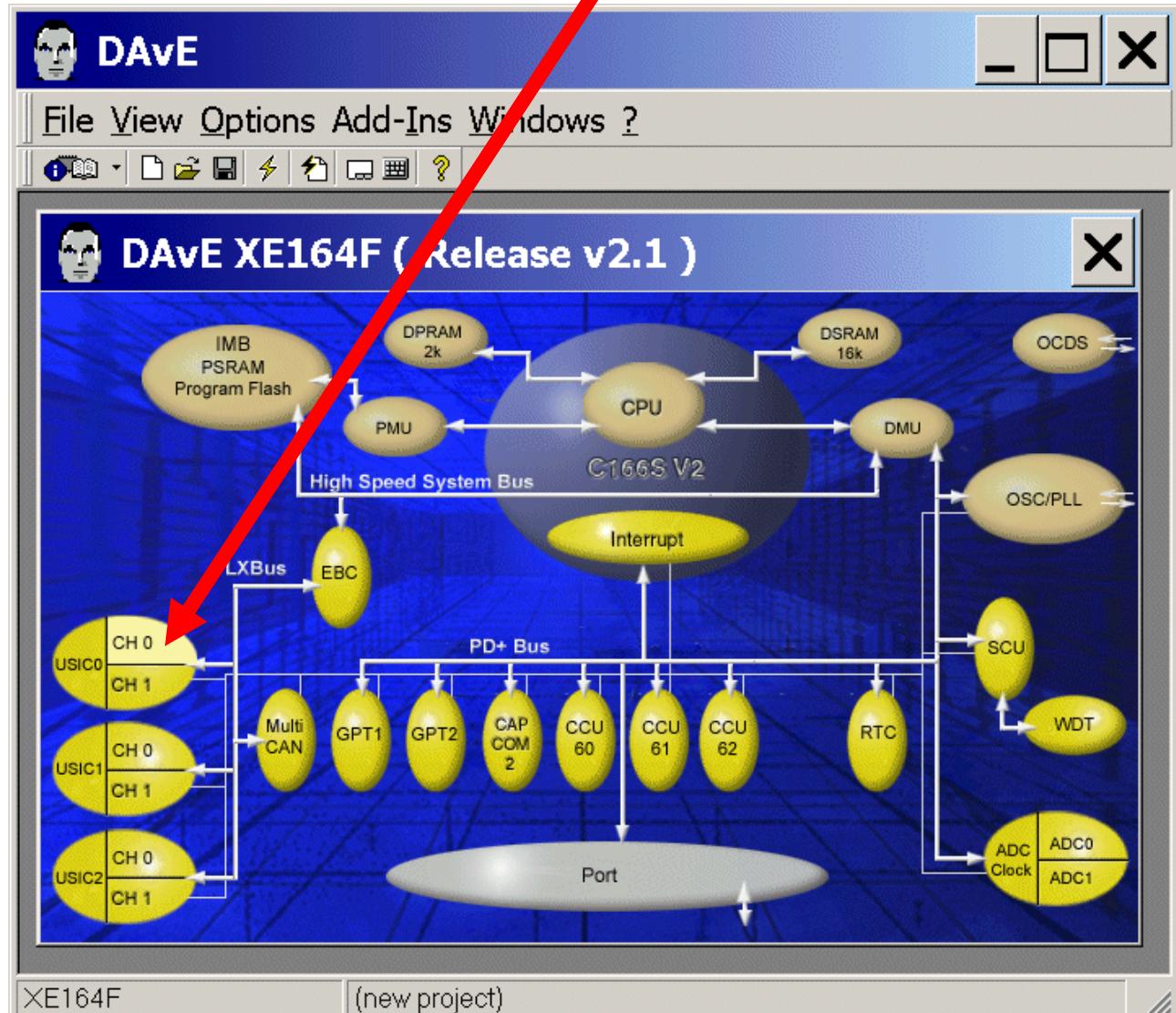
Protocol: USIC0 Channel 0 Protocol Selection: USIC Protocol for U0C0: select ASC



Exit and Save this dialog now by clicking  the close button.

Configuration of the serial interface **USIC0\_CH0 / U0C0** (**Universal Serial Interface Channel module 0, Channel 0**):

The configuration window/dialog can be opened by clicking the specific block/module (CH 0).



**Note:**

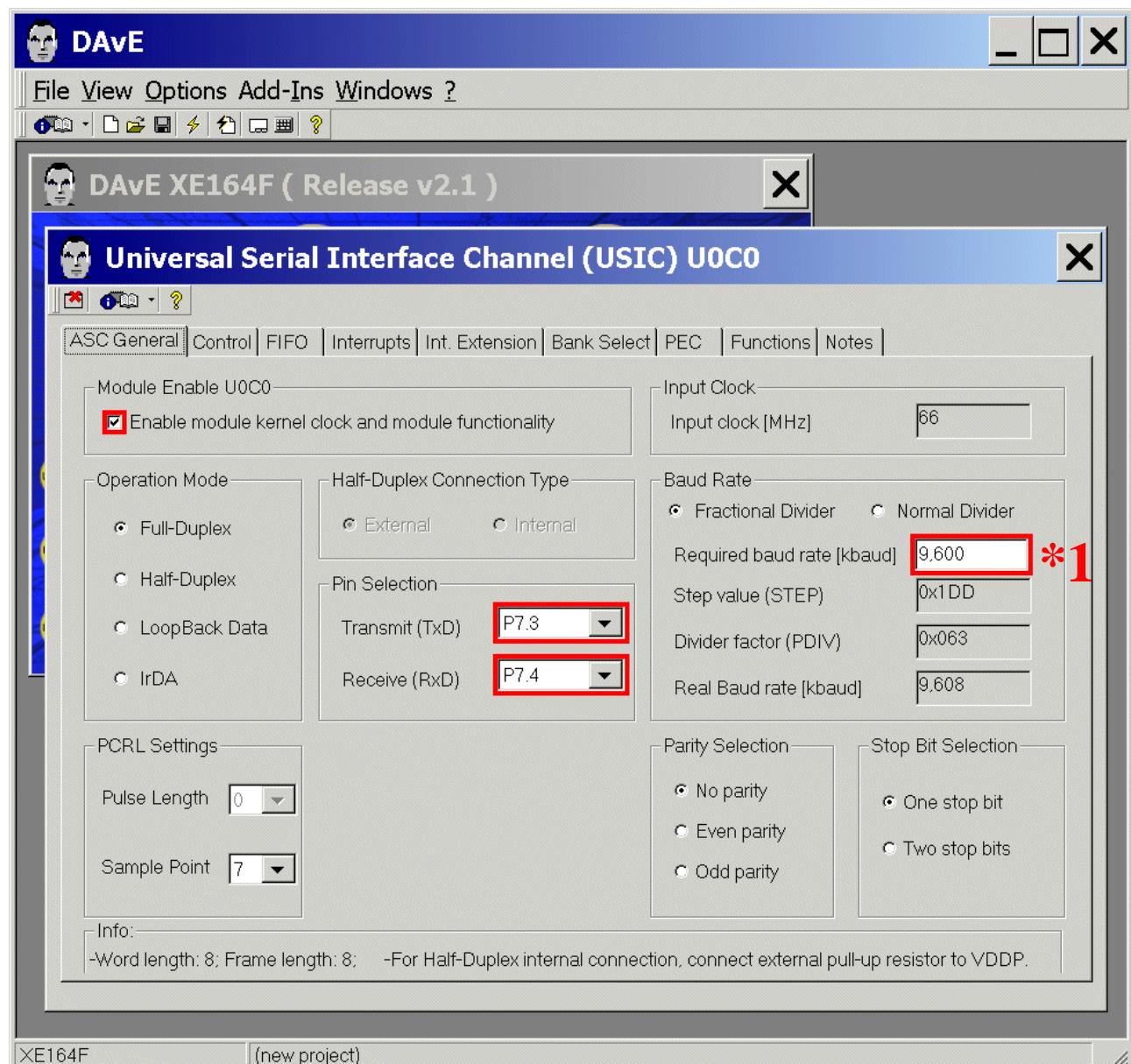
For serial communication with a terminal program running on your host computer U0C0 is used.

ASC General: Module Enable U0C0: tick  Enable module kernel clock and module functionality

ASC General: Pin Selection: Transmit (TxD): select P7.3

ASC General: Pin Selection: Receive (RxD): select P7.4

ASC General: Baud Rate: Required baud rate [kbaud]: insert 9,600 <ENTER>



#### Note (\*1):

Validate each alphanumeric entry by pressing <ENTER>.

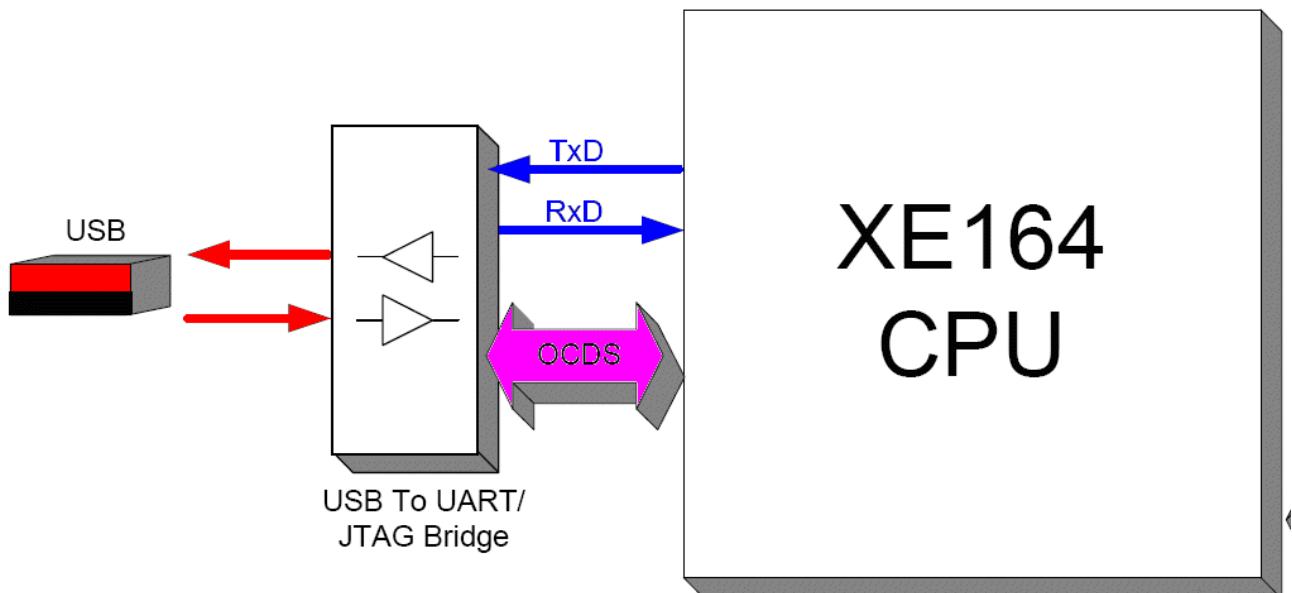




Additional information: [RS232 serial interface](#):

**Note:**

The RS232 serial interface ([USIC\\_0\\_Channel\\_0 pins P7.3 and P7.4](#)) is available via the [USB port](#) which converts the TTL-UART-signals to USB-signals (using a virtual COM port of the second USB channel of the FTDI FT2232 Dual USB to UART/JTAG interface).





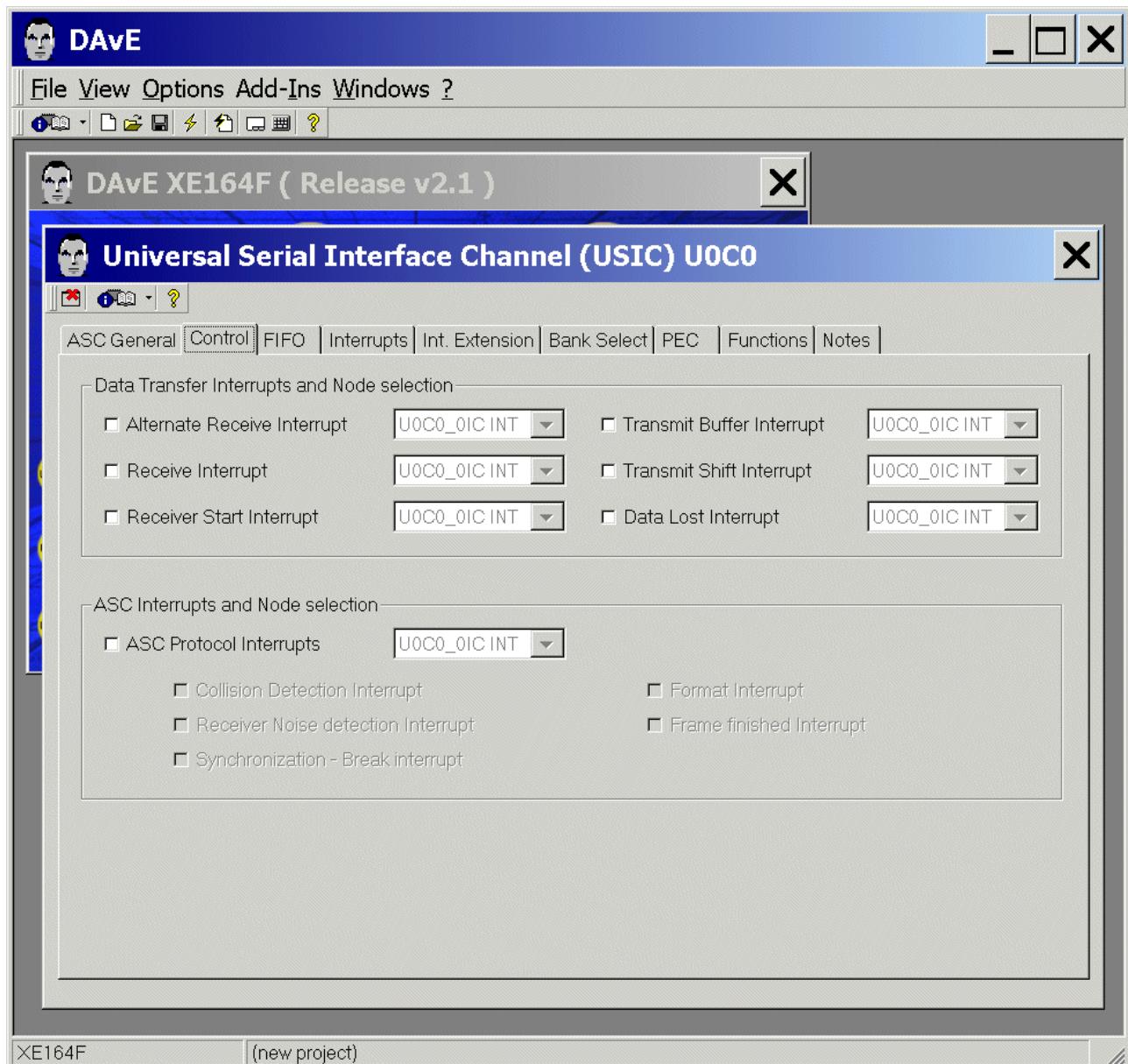
Additional information: [Standard UART / USIC\\_0\\_Channel\\_0 Pins](#) (Source: User's Manual):

**Table 10-10 Configuration Data for Bootstrap Loader Modes**

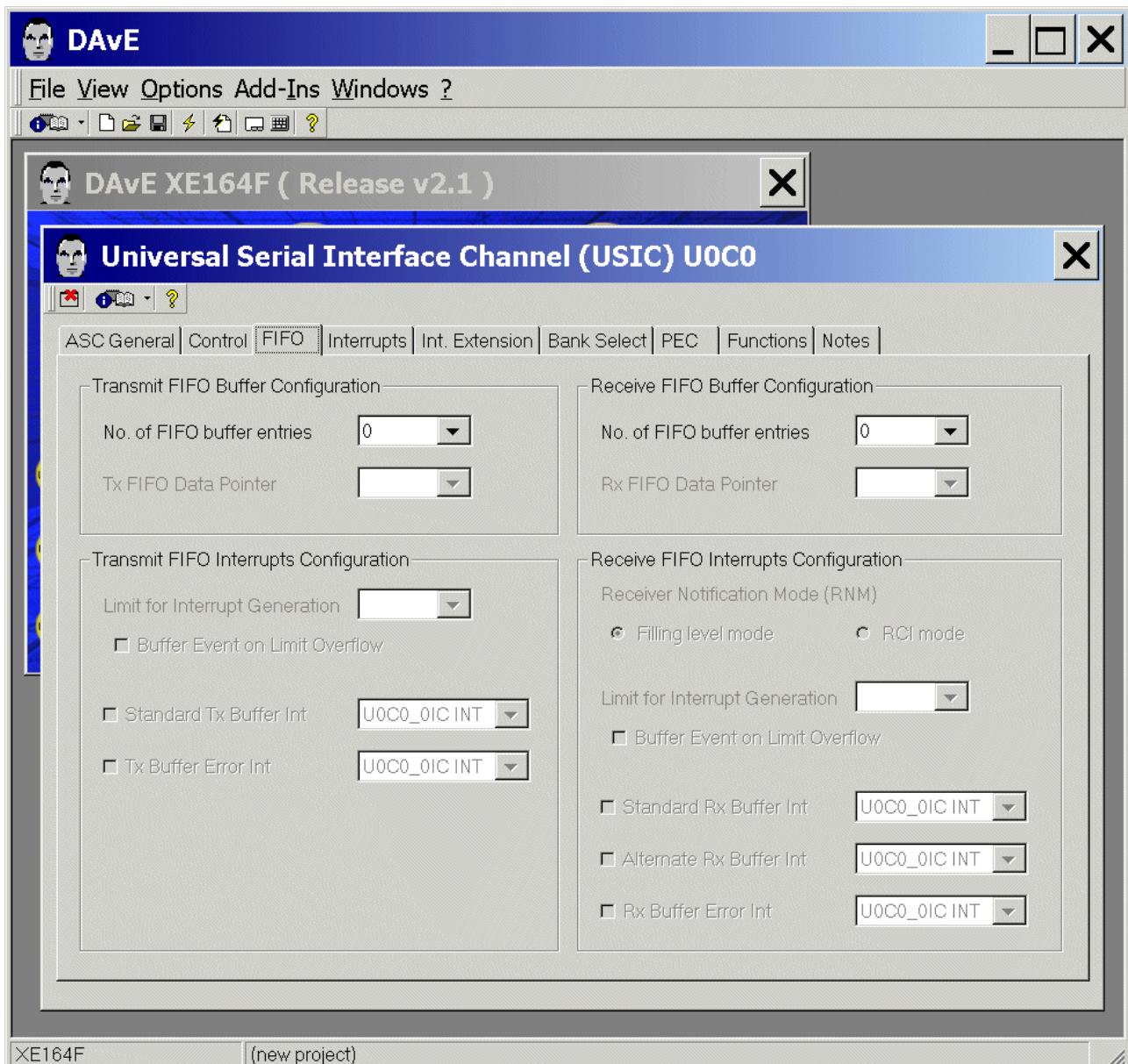
Bootstrap Loader Mode	Configuration on P10.3-0 <sup>1)</sup>	Receive Line from Host	Transmit Line to Host	Transferred Data
Standard UART	$x110_B$	RxD = P7.4	TxD = P7.3	32 Bytes
Sync. Serial	$1001_B$	MRST = P2.4	MTSR = P2.3 SCLK = P2.5 SLS = P2.6	n Bytes; 1 ... 65,280
MultiCAN	$x101_B$	RxDC0 = P2.6	TxDC0 = P2.5	$8 \times n$ Bytes

1) x means that the level on the corresponding pin is irrelevant.

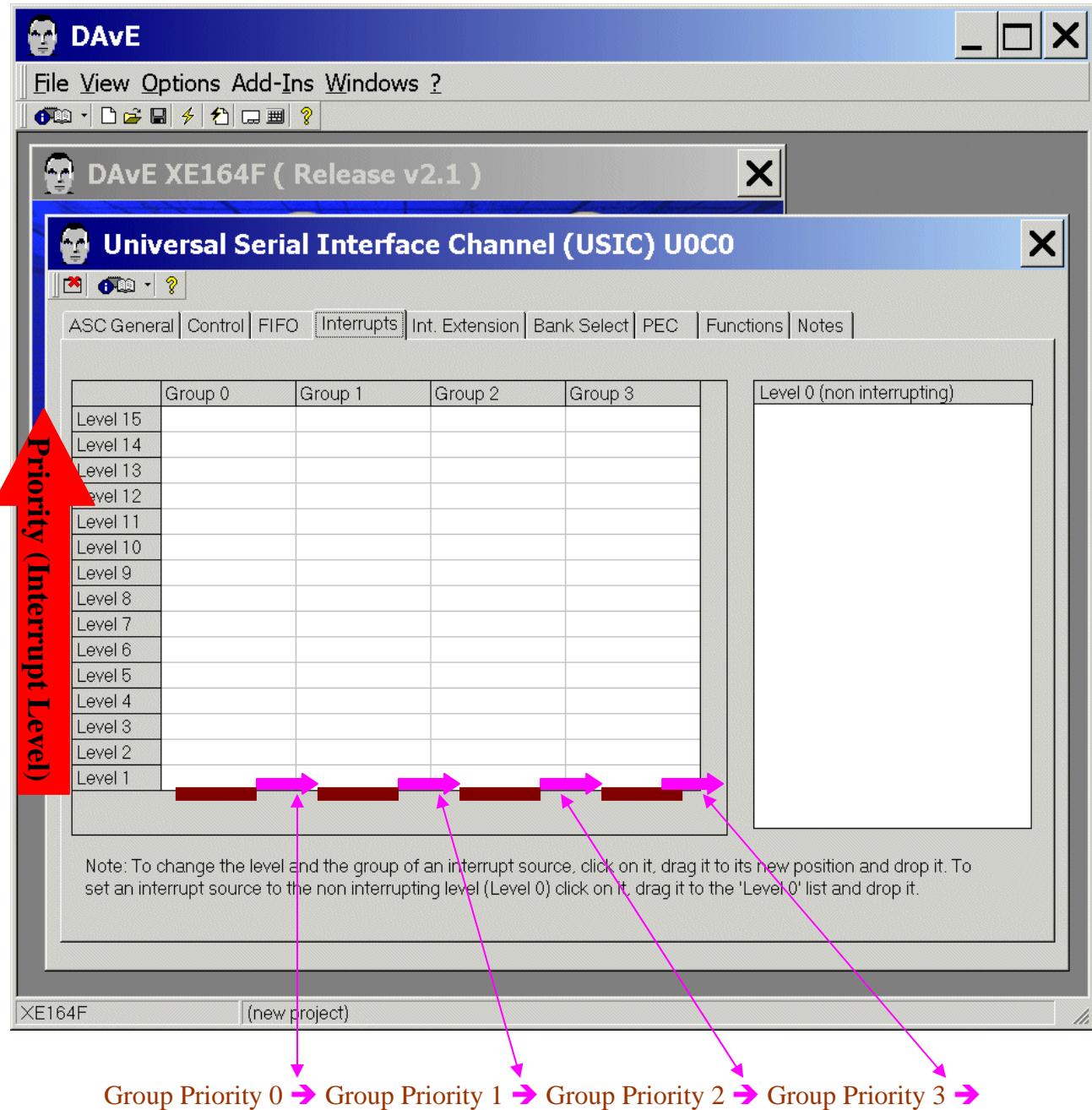
**Control:** (do nothing)



**FIFO:** (do nothing)



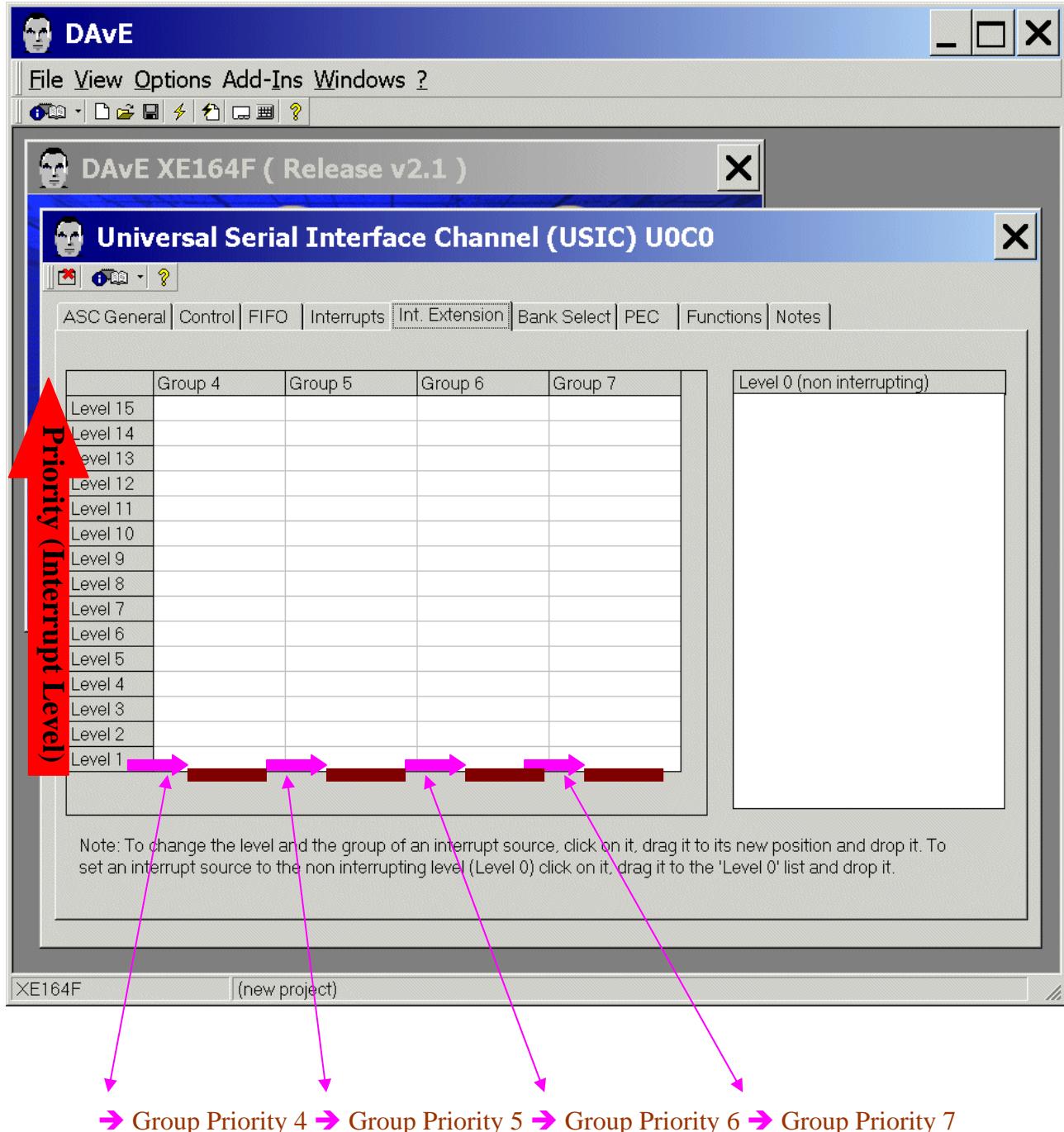
Interrupts: (do nothing)



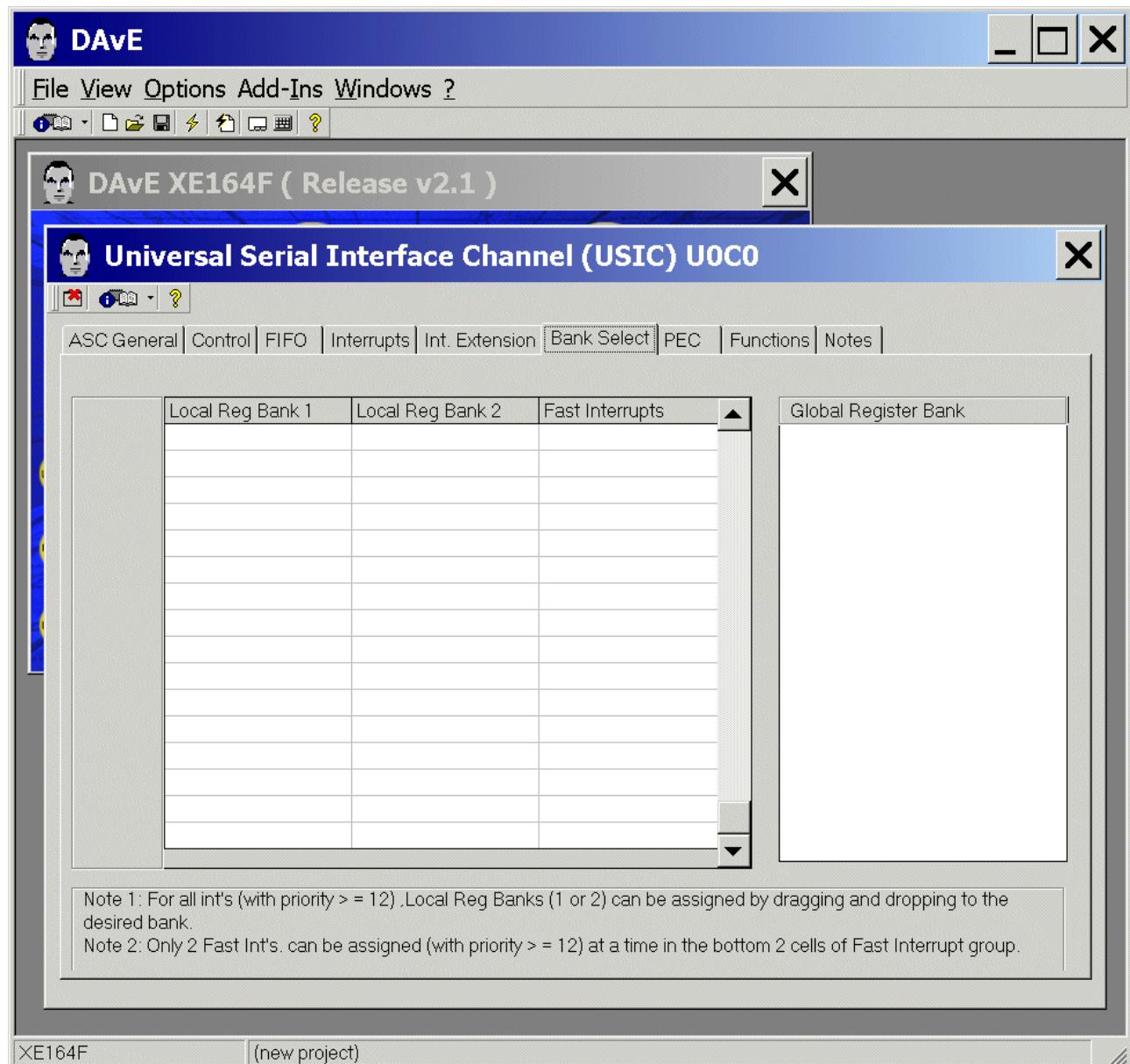
**Note:**

For the serial communication with a terminal program (e.g. Docklight, [www.docklight.de](http://www.docklight.de)) running on your host computer the myprintf function is used. The myprintf function uses Software-Polling-Mode therefore we do not need to configure any interrupts.

Int. Extension: (do nothing)

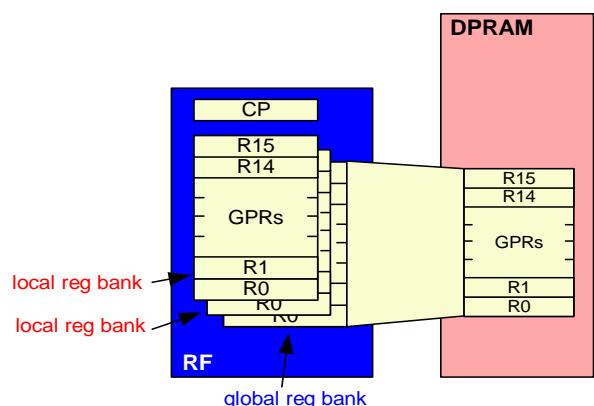


**Bank Select:** (do nothing)



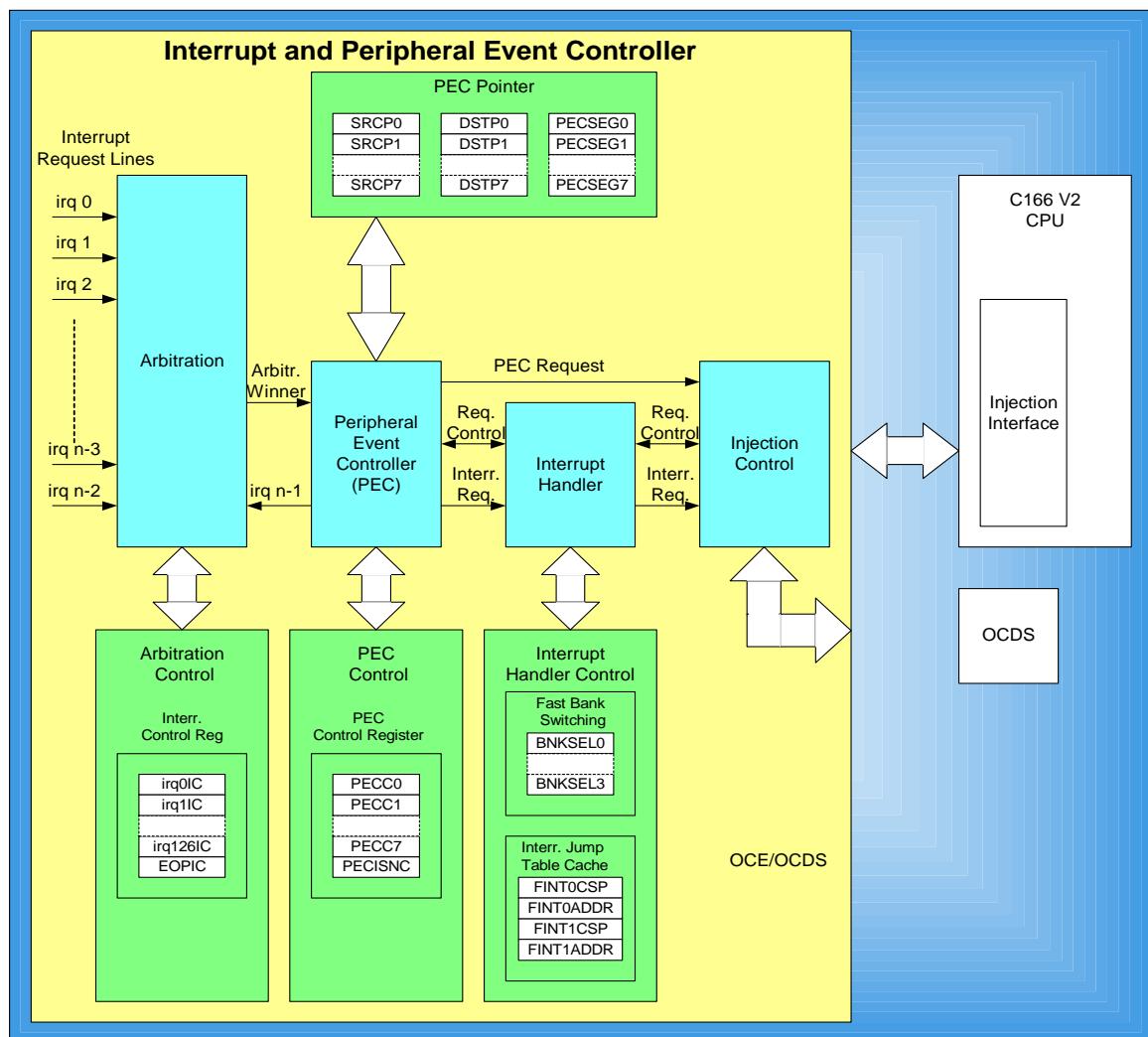
**Note:**

For our hello world program the 2 local register banks are not needed.





Additional information: **local register banks** and **fast interrupts** (page 1/2):





Additional information: **local register banks** and **fast interrupts** (page 1/2):

```
void CC1_viCC1 (void) interrupt CC1_CC1INT { }

void CC1_viCC1 (void) interrupt CC1_CC1INT using REGBANK4711 { }

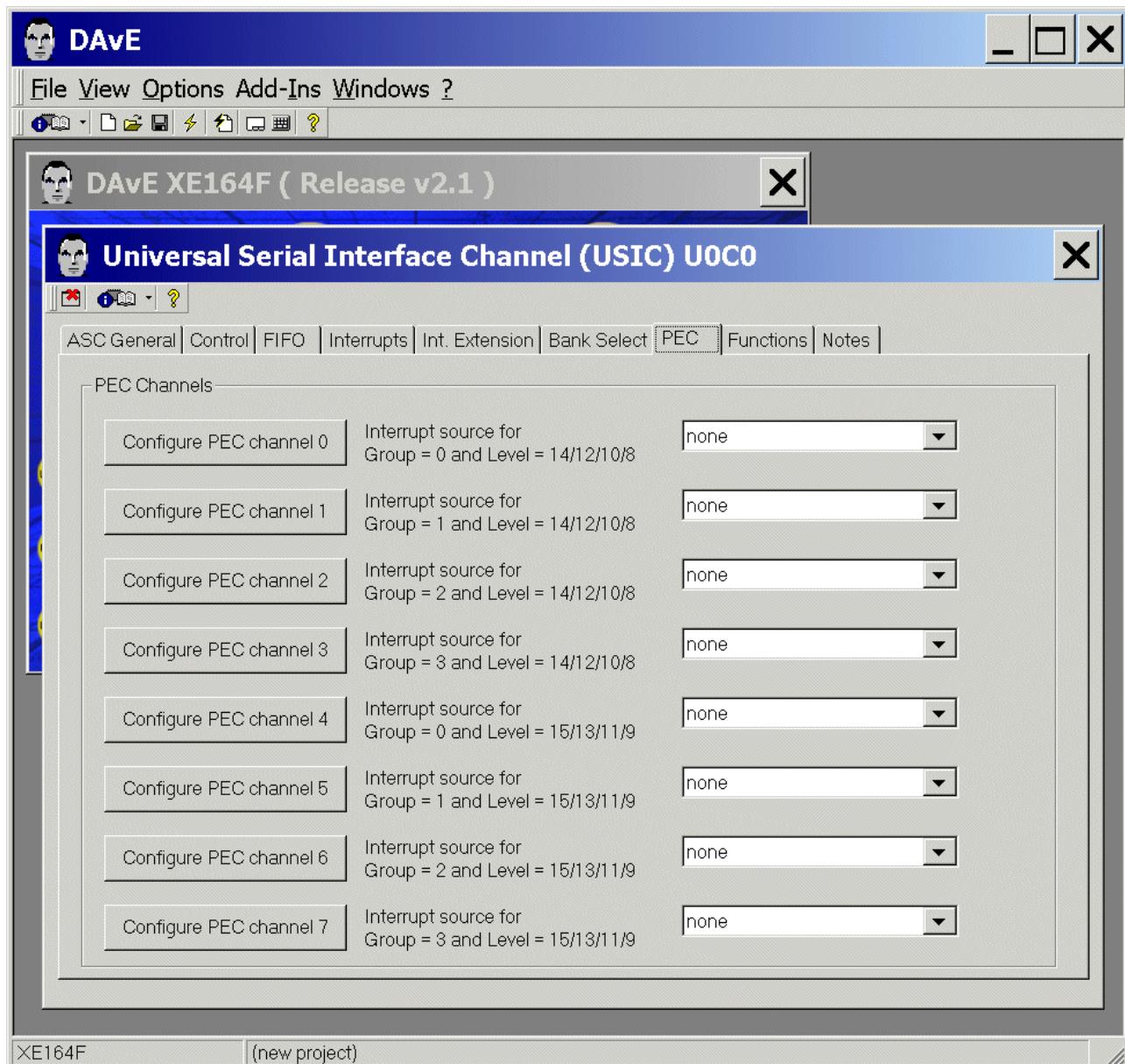
void CC1_viCC1 (void) interrupt CC1_CC1INT using _FAST_ABANK1_ { }

void CC1_viCC1 (void) interrupt MickeyMouse=CACHED { }

void CC1_viCC1 (void) interrupt MickeyMouse=CACHED using REGBANK4711 { }

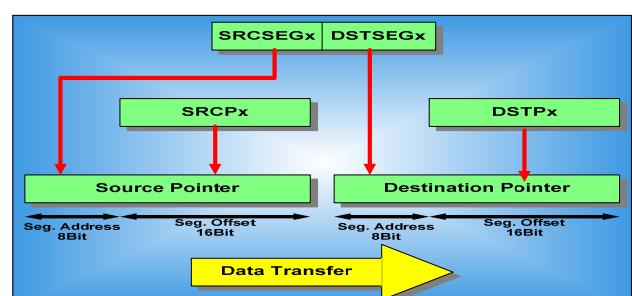
void CC1_viCC1 (void) interrupt MickeyMouse=CACHED using _FAST_ABANK1_ { }
```

**PEC:** (do nothing)

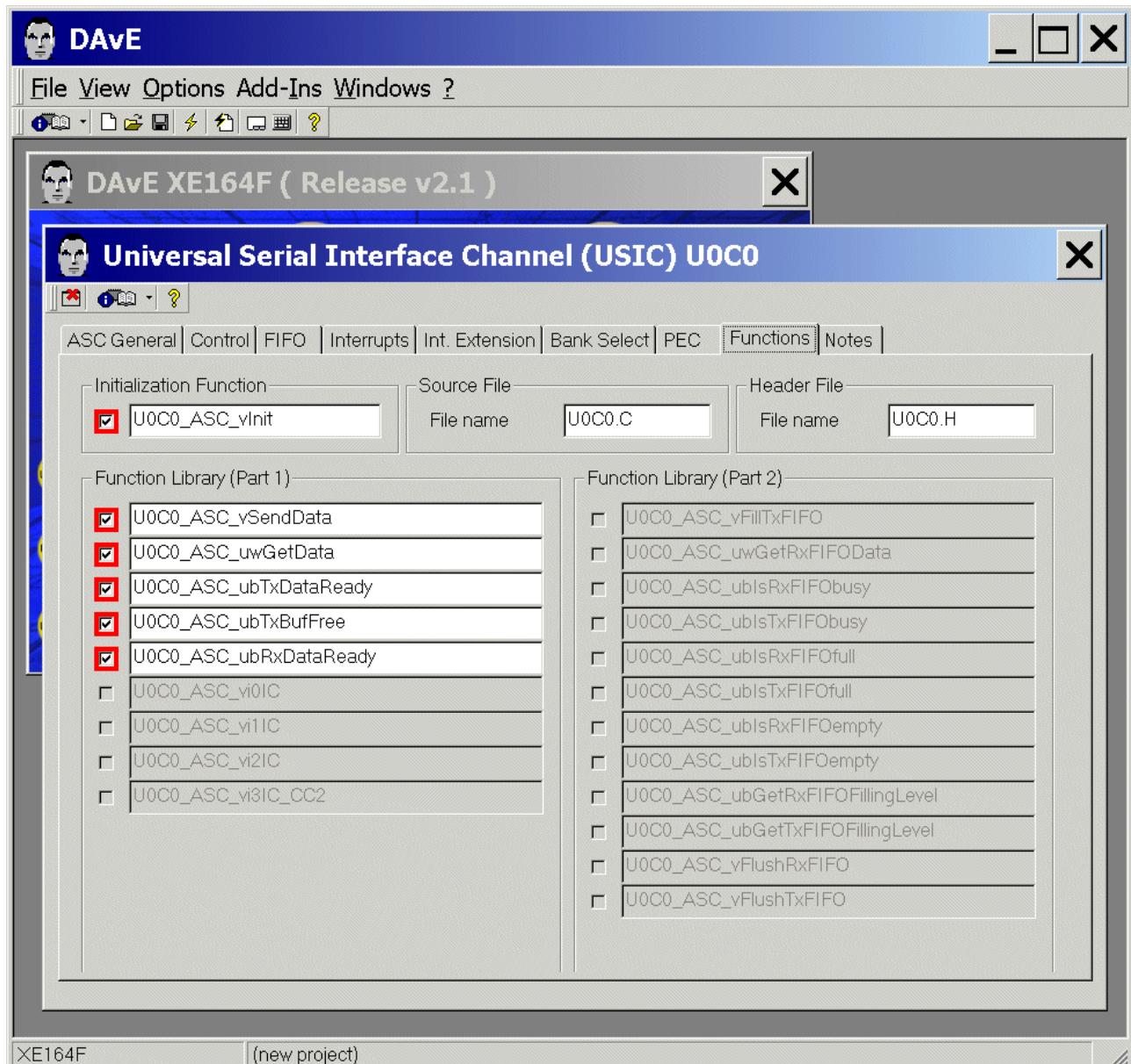


**Note:**

For our hello world program the 8 PEC Channels are not needed.



Functions: Initialization Function: tick/check  U0C0\_ASC\_vInit  
 Functions: Function Library (Part 1): tick  U0C0\_ASC\_vSendData  
 Functions: Function Library (Part 1): tick  U0C0\_ASC\_uwGetData  
 Functions: Function Library (Part 1): tick  U0C0\_ASC\_ubTxDataReady  
 Functions: Function Library (Part 1): tick  U0C0\_ASC\_ubTxBufFree  
 Functions: Function Library (Part 1): tick  U0C0\_ASC\_ubRxDataReady

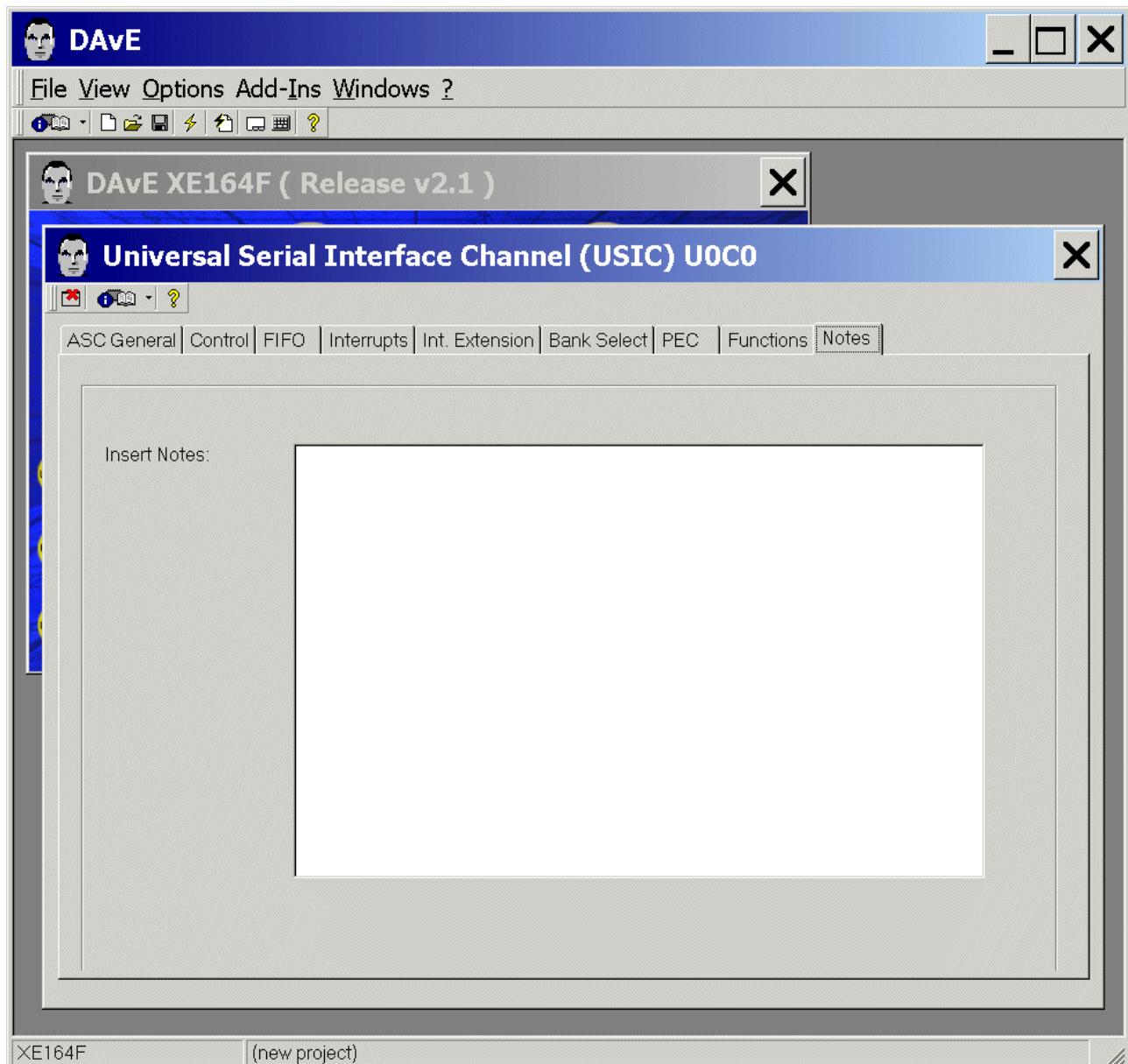


#### Note:

You can change function names (e.g. U0C0\_ASC\_vInit) and file names (e.g. U0C0.C, U0C0.H) anytime.



Notes: (do nothing)



Note:

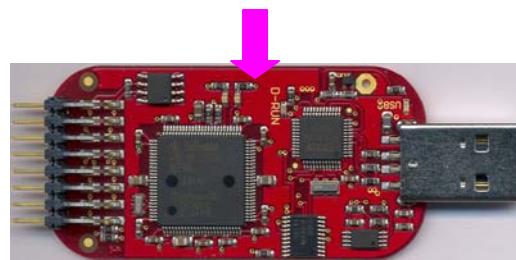
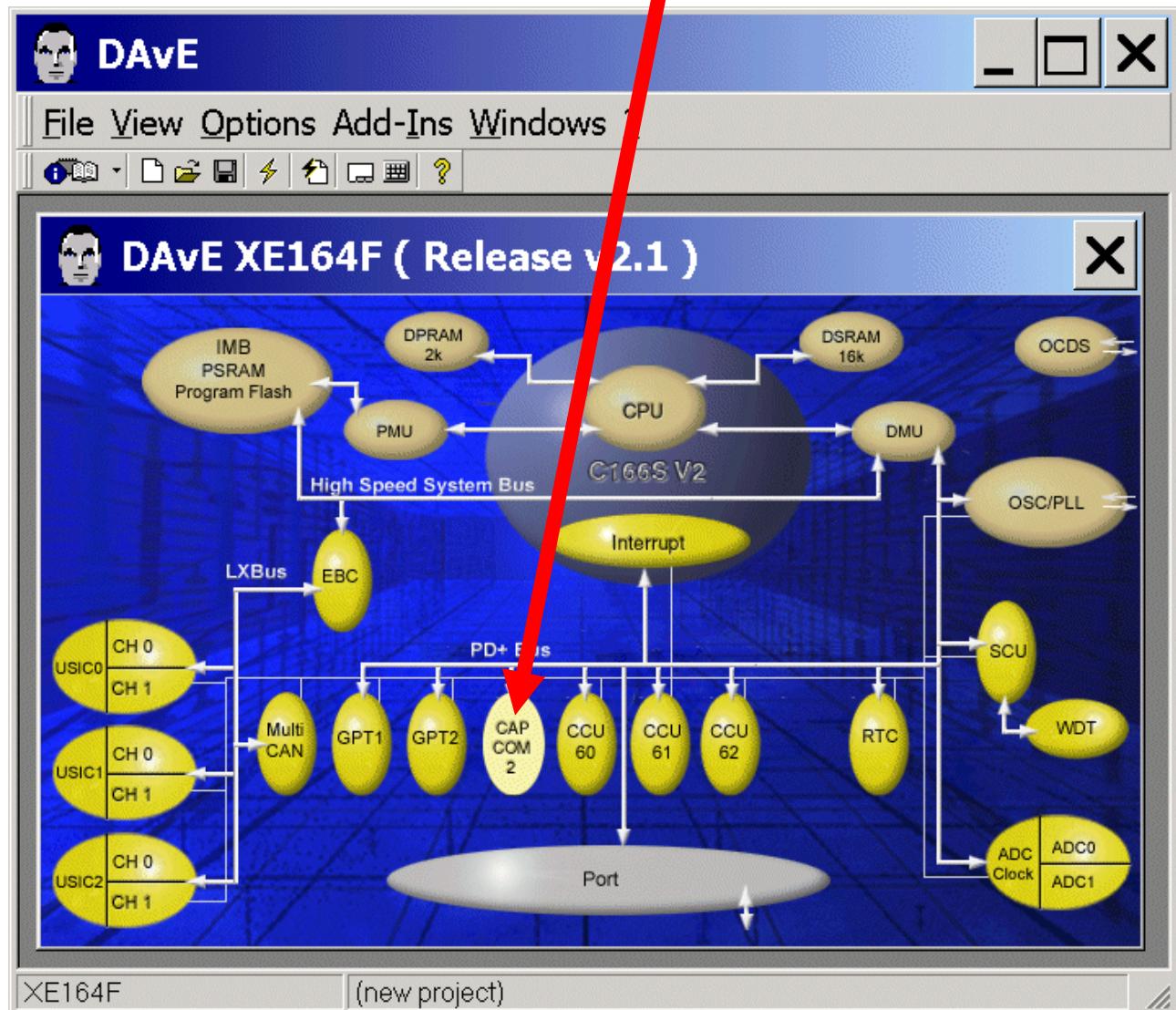
Notes: Insert Notes: If you wish, you can insert your comments here.



Exit and Save this dialog now by clicking  the close button.

Configure Timer T7 in the CAPCOM 2 module:

The configuration window/dialog can be opened by clicking the specific block/module (CAPCOM2).



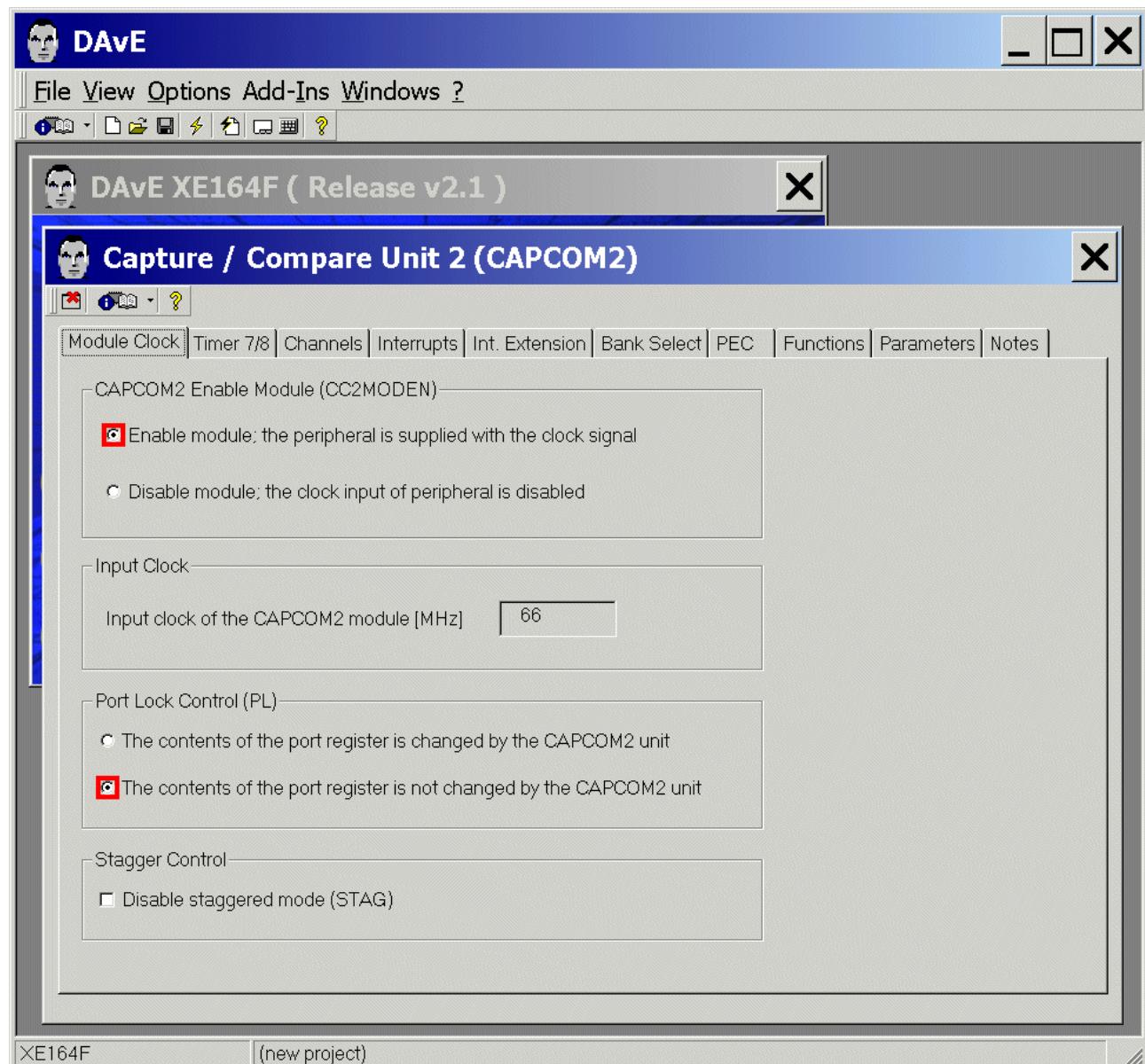
**Note:**

The LED on IO\_Port\_2.7 will be blinking (after Power-On or if selected in the main menu) with a frequency of about 1 second (done in the Timer\_7-Interrupt-Service-Routine). Therefore we have to configure Timer\_7.

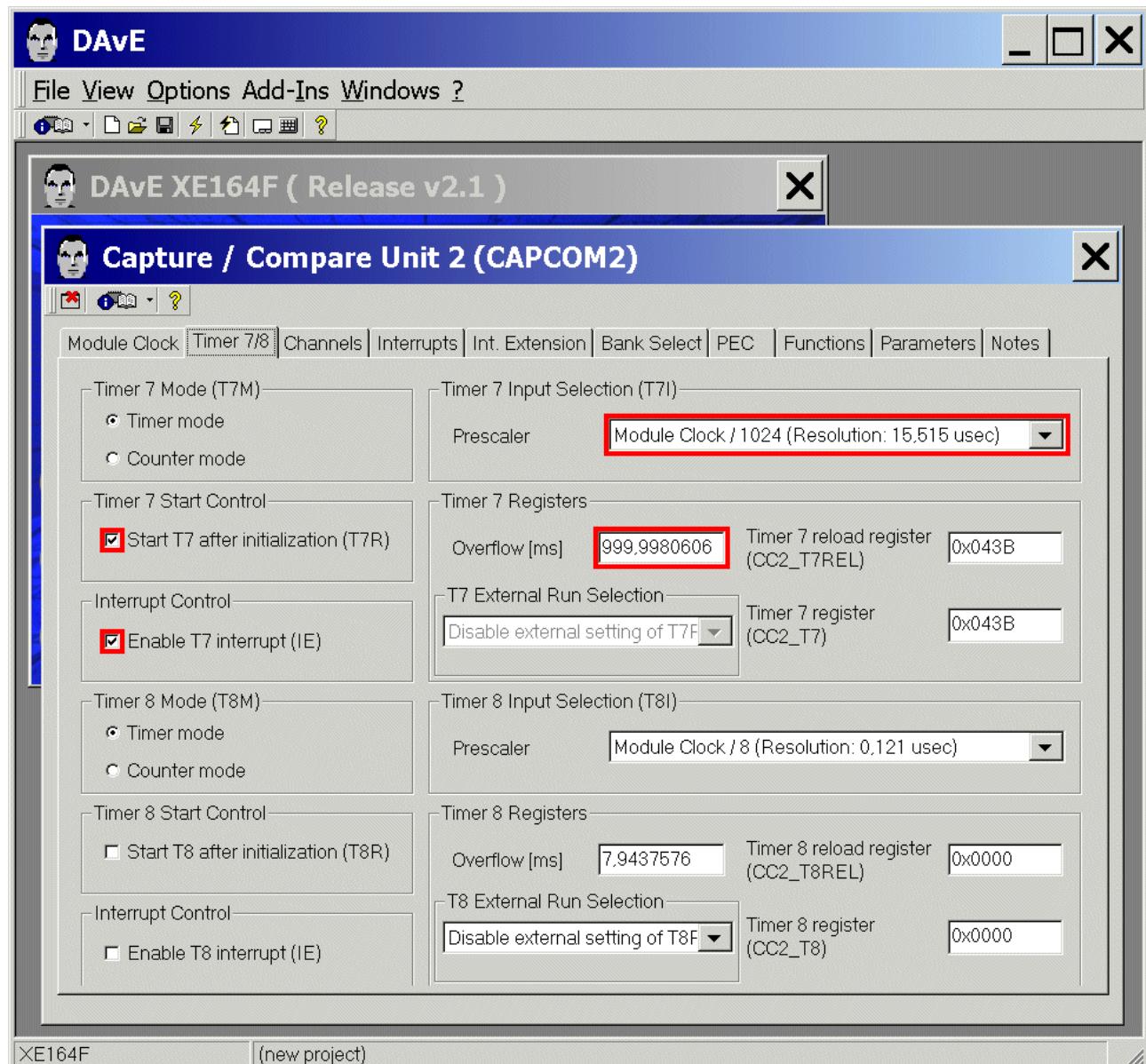
Module Clock: CAPCOM2 Enable Module: **click**  Enable module

Module Clock: Port Lock Control:

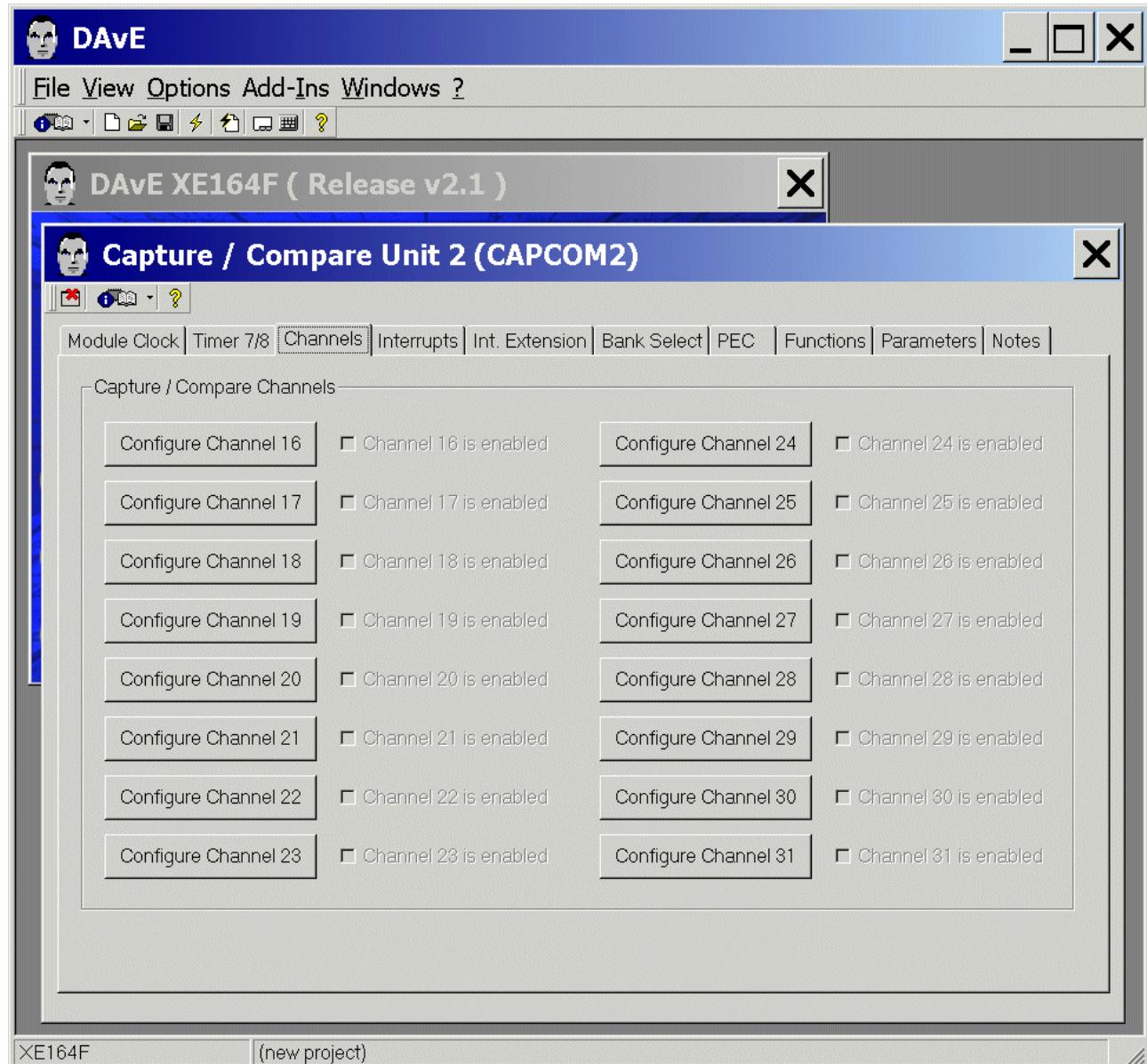
**click**  The contents of the port register is not changed by the CAPCOM2 unit



Timer 7/8: Timer 7 Start Control: **tick ✓ Start T7 after initialization (T7R)**  
 Timer 7/8: Interrupt Control: **tick ✓ Enable T7 interrupt (IE)**  
 Timer 7/8: Timer 7 Input Selection (T7I): Prescaler: **choose Module Clock/1024**  
 Timer 7/8: Timer 7 Registers: Overflow [s]: **insert 1 <ENTER>**

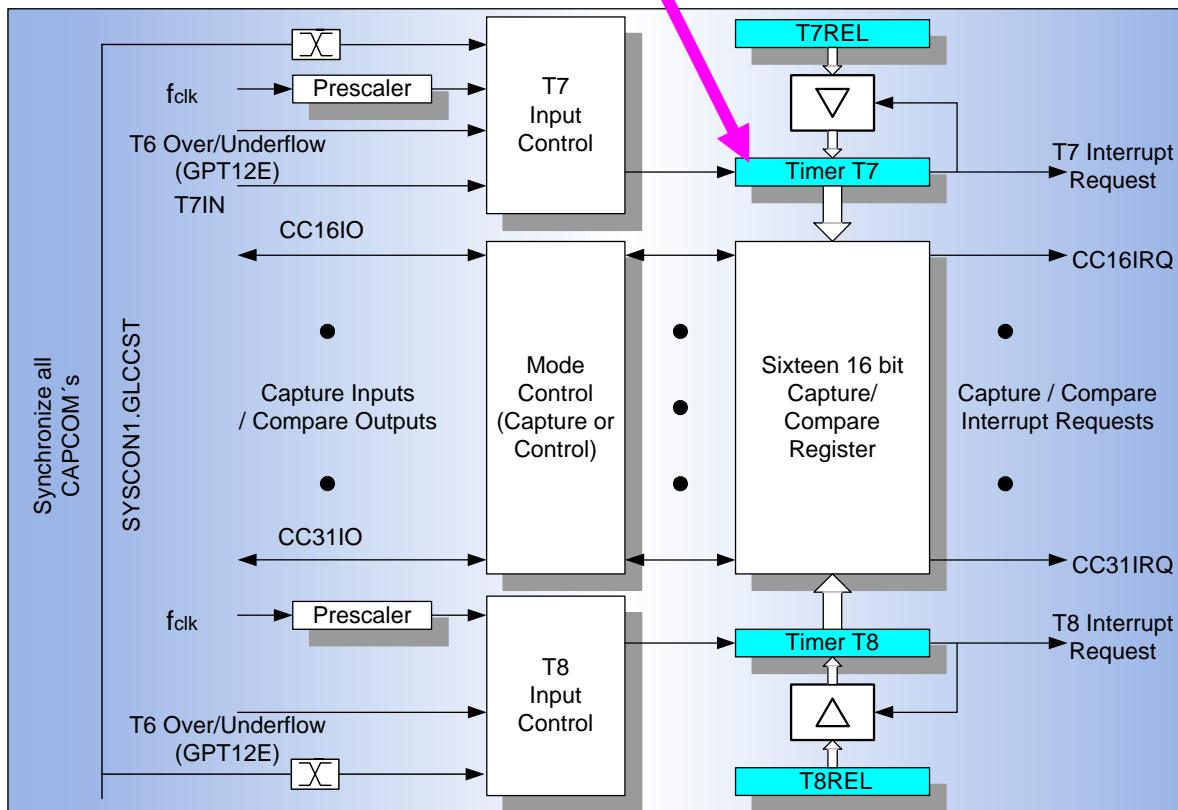


**Channels:** (do nothing)

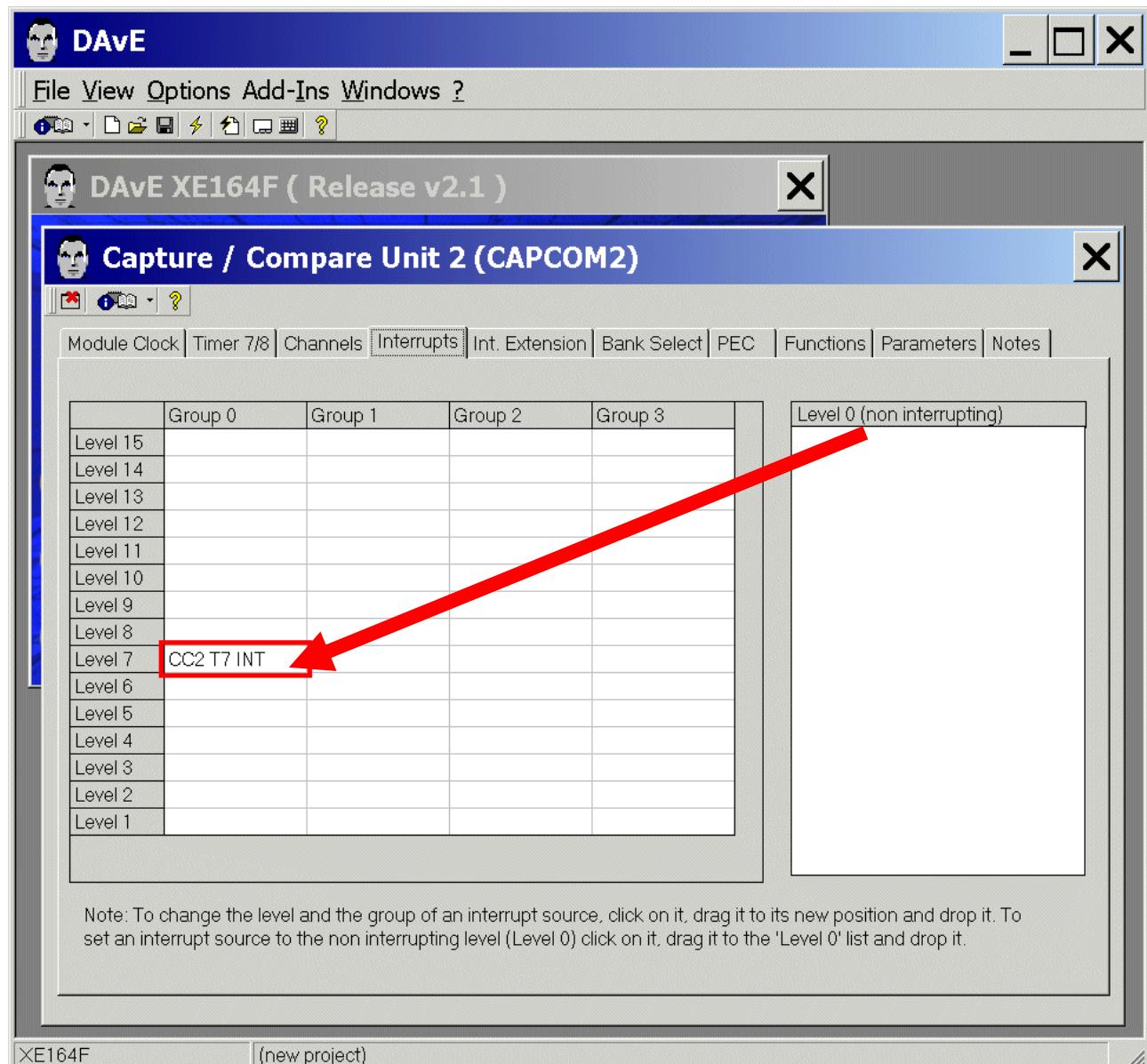



**Note:**

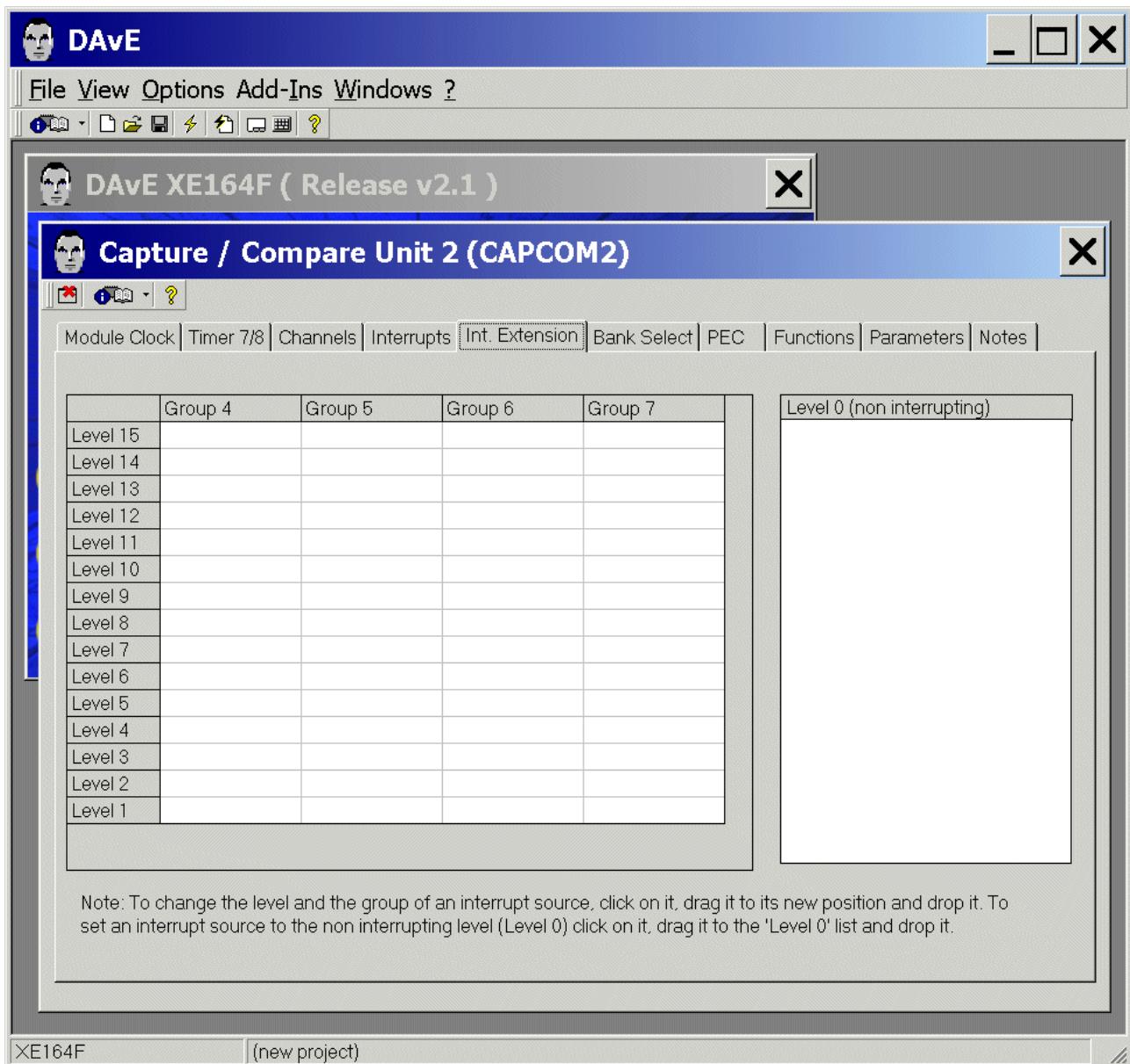
For our hello world program only **Timer 7** in the CAPCOM 2 module is used.



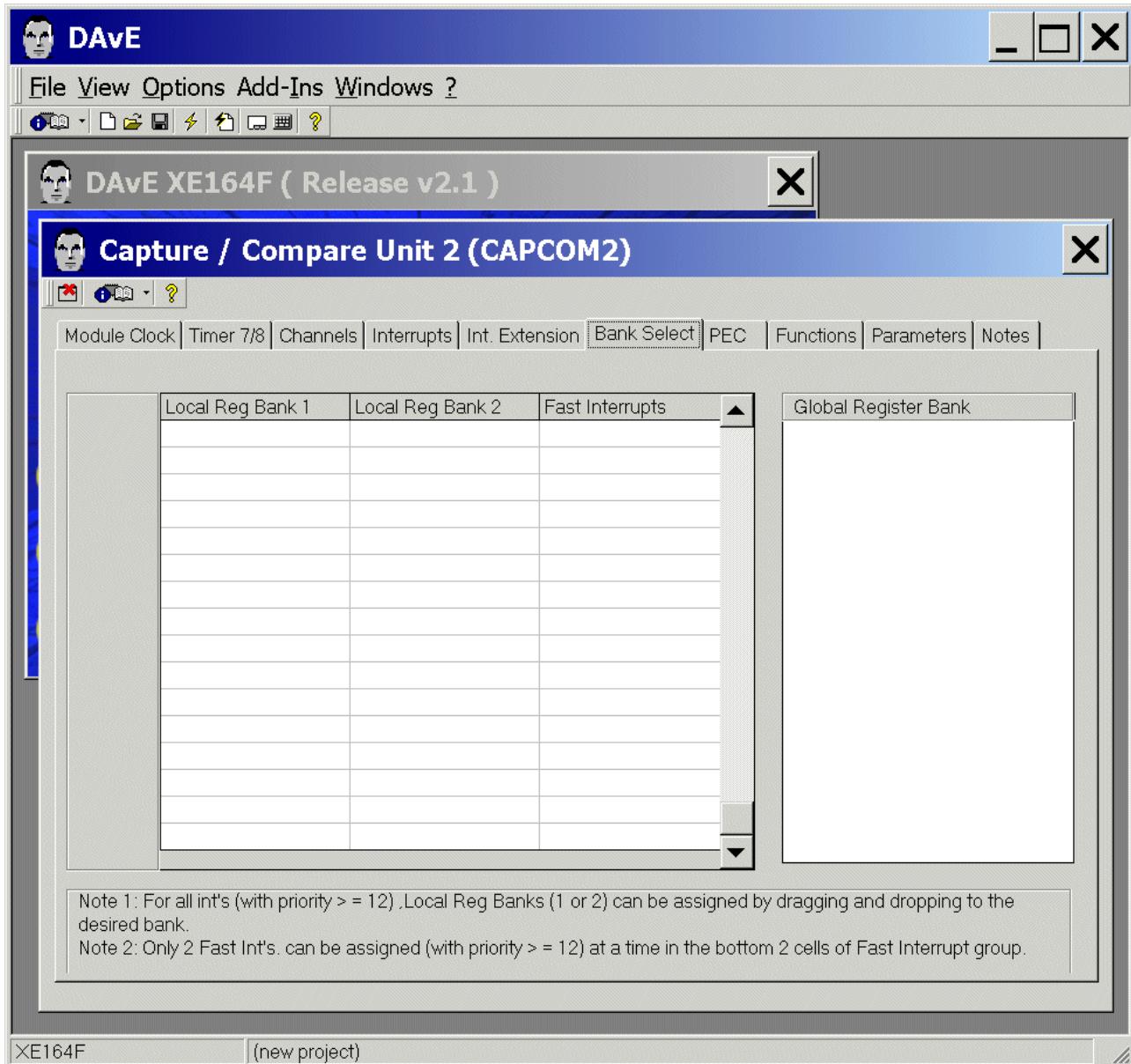
Interrupts: drag and drop the CC2 T7 INT to Interrupt Level 7, Group 0



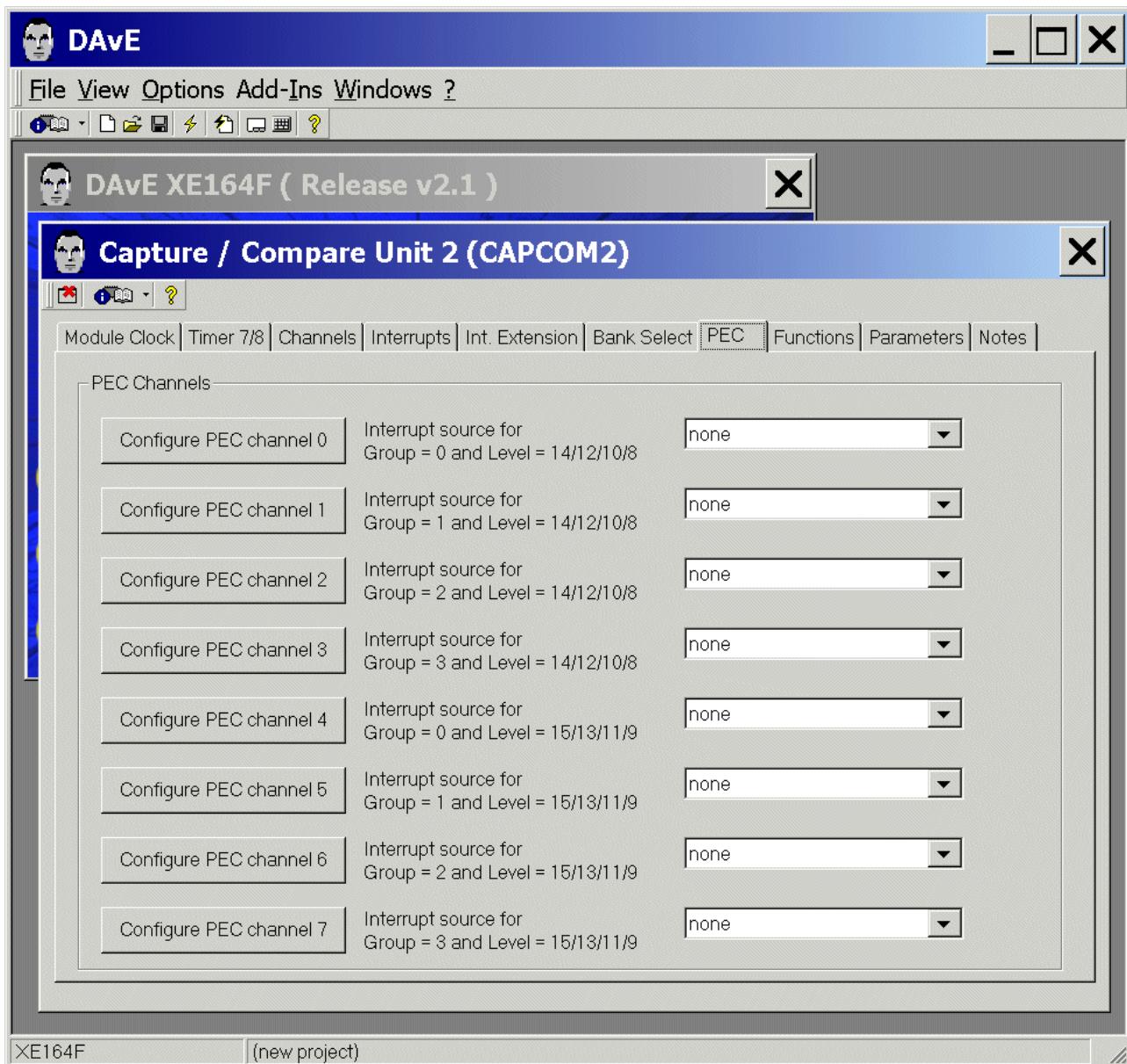
Int. Extension: (do nothing)



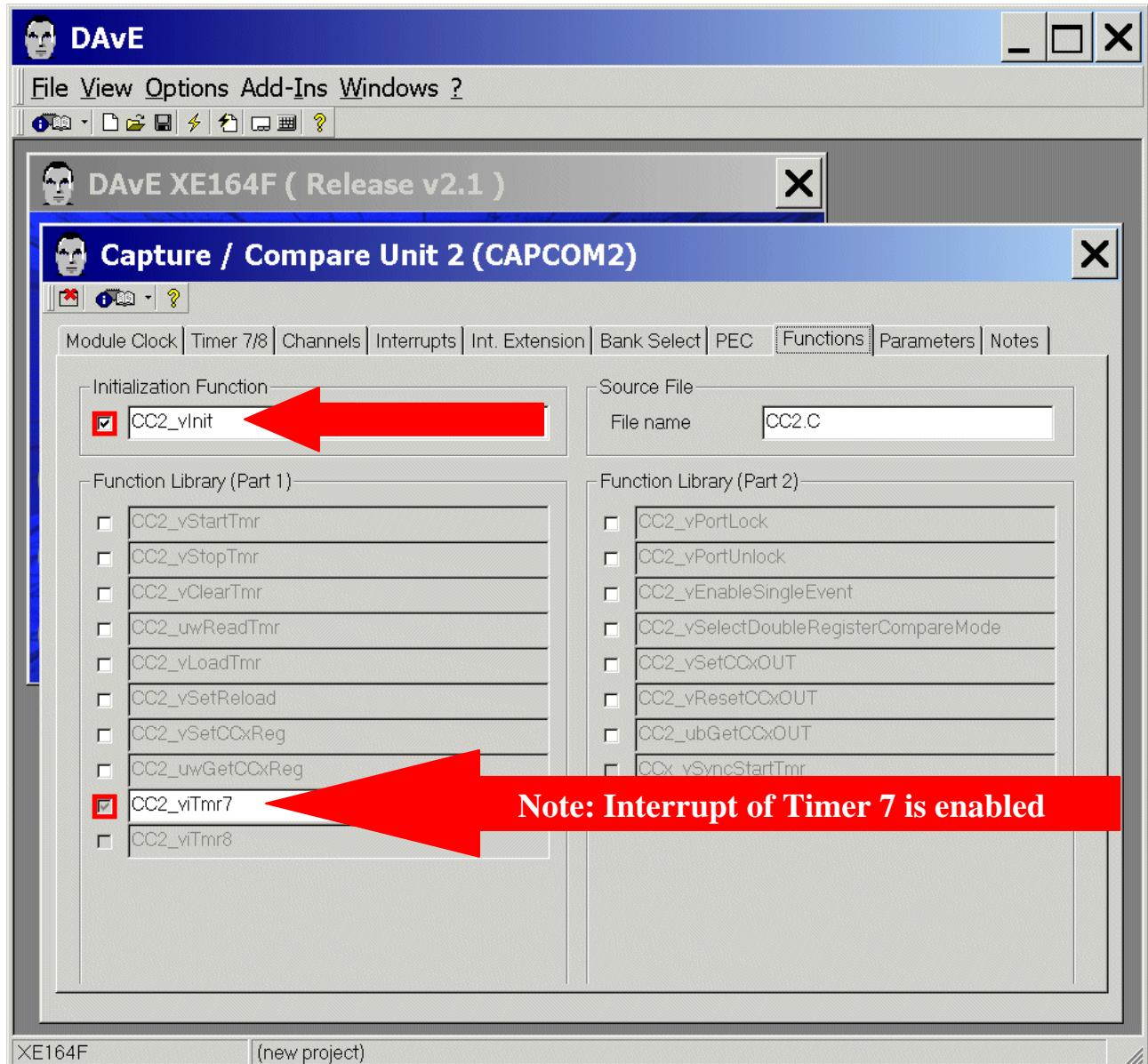
Bank Select: (do nothing)



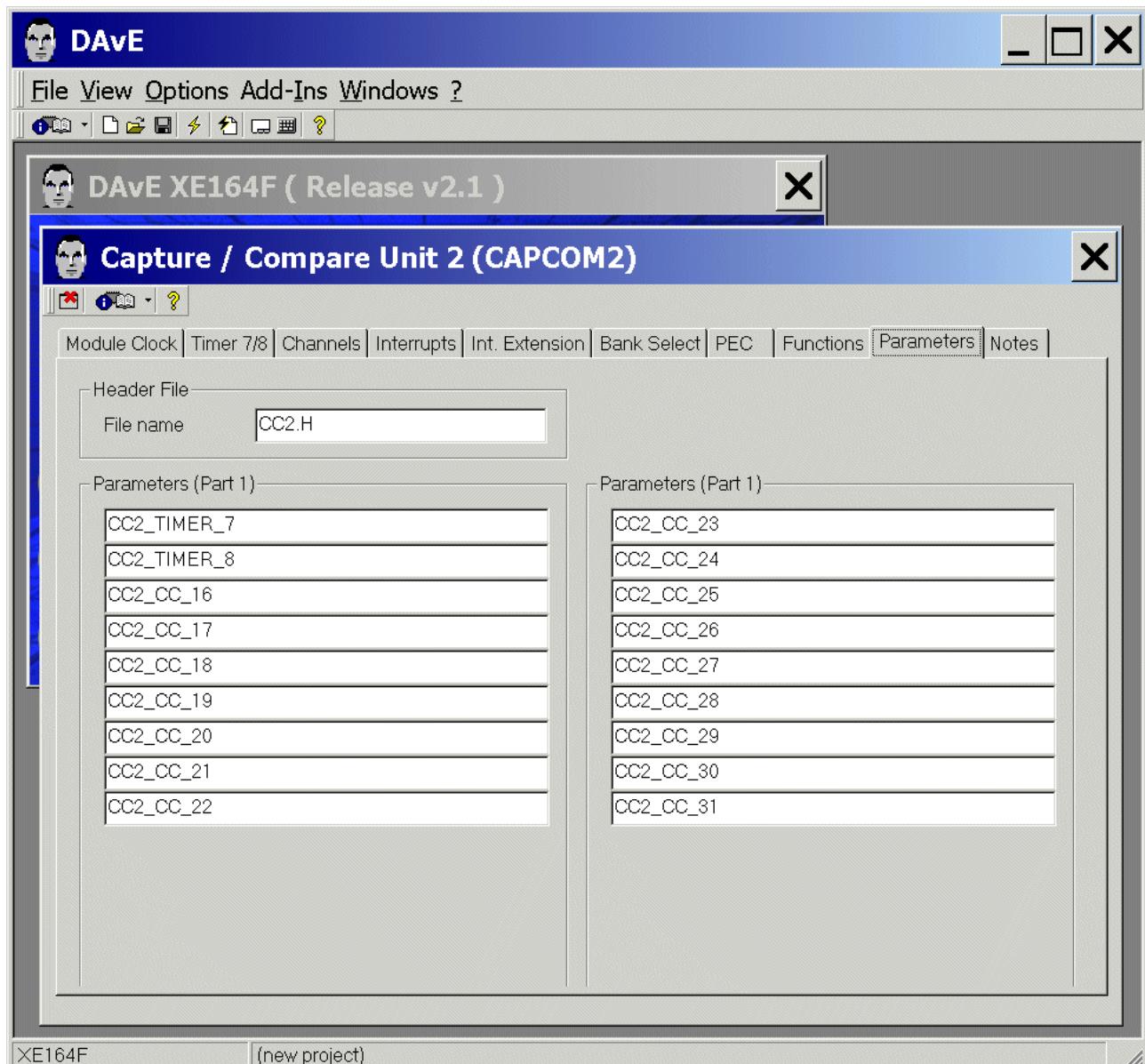
**PEC:** (do nothing)



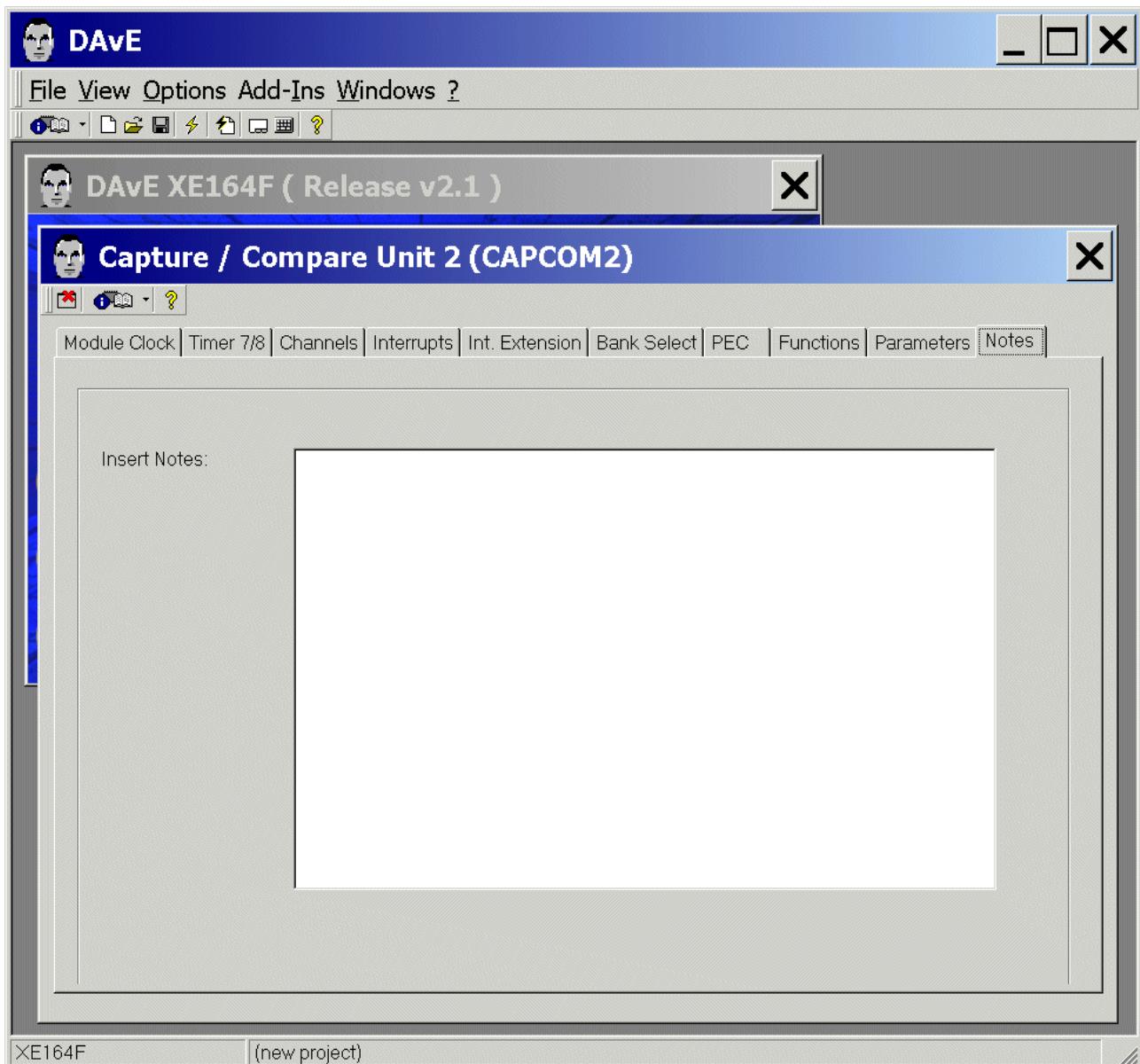
Functions: Initialization Function: tick/check  CC2\_vInit



Parameters: (do nothing)



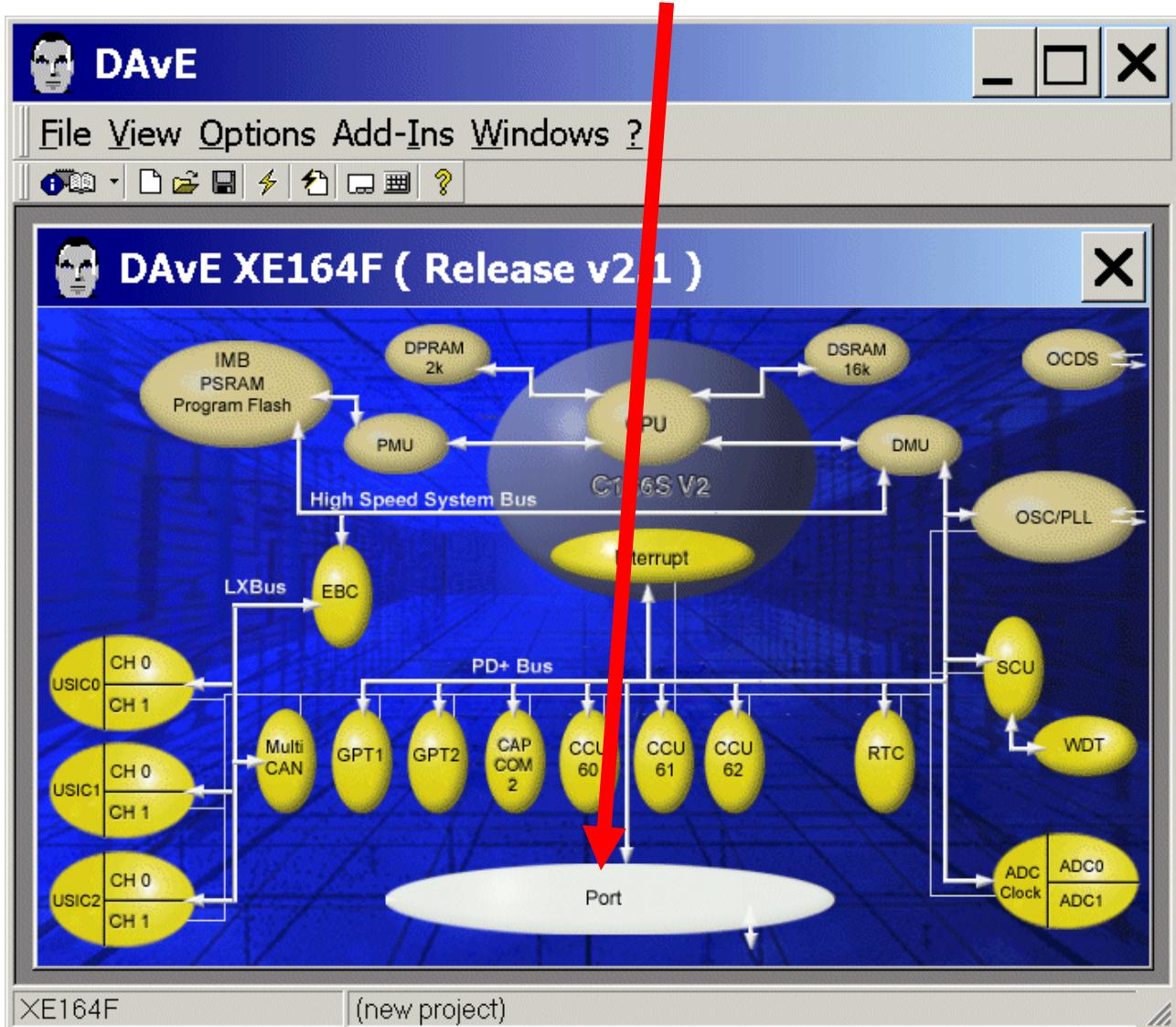
**Notes:** Insert Notes: If you wish, you can insert your comments here.



Exit and Save this dialog now by clicking  the close button.

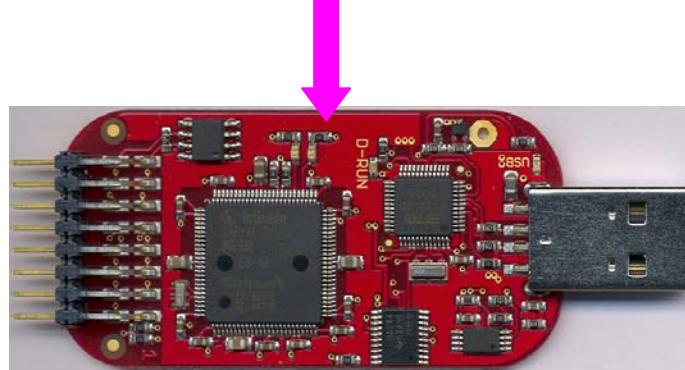
Configure Port 2 Pin 7 to Output :

The configuration window/dialog can be opened by clicking the specific block/module (Port).

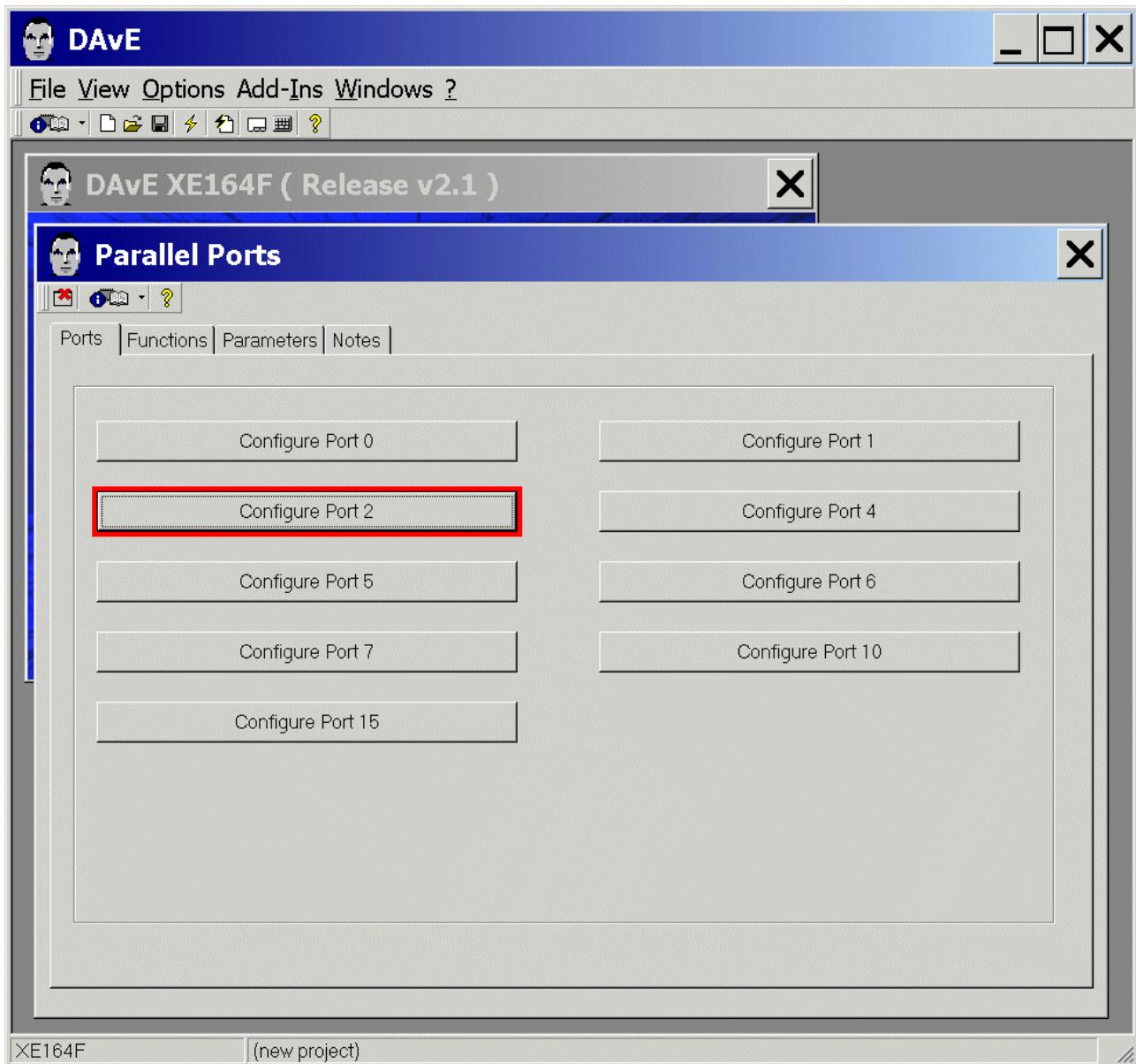


Note:

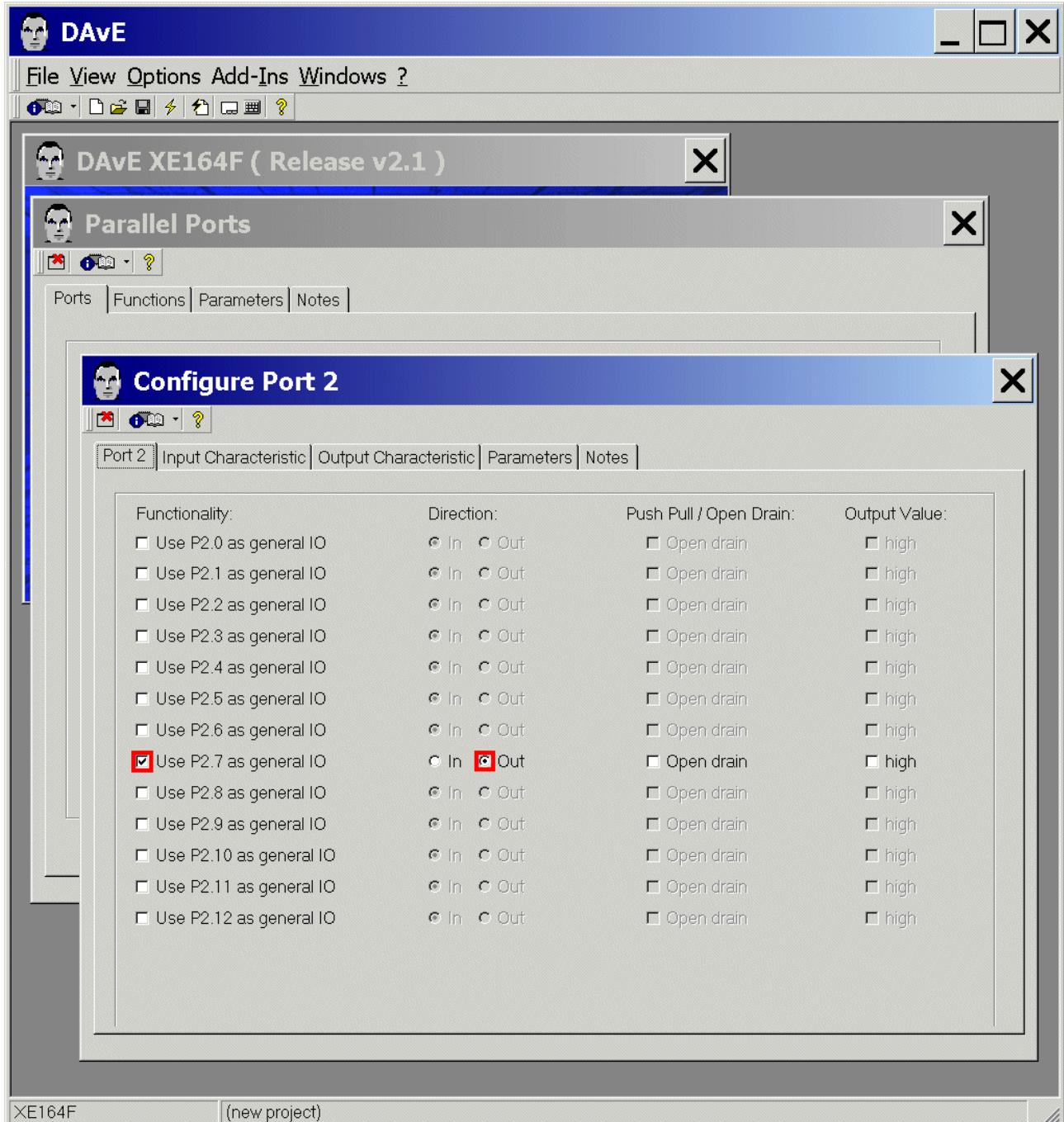
The LED is connected to IO\_Port\_2.7



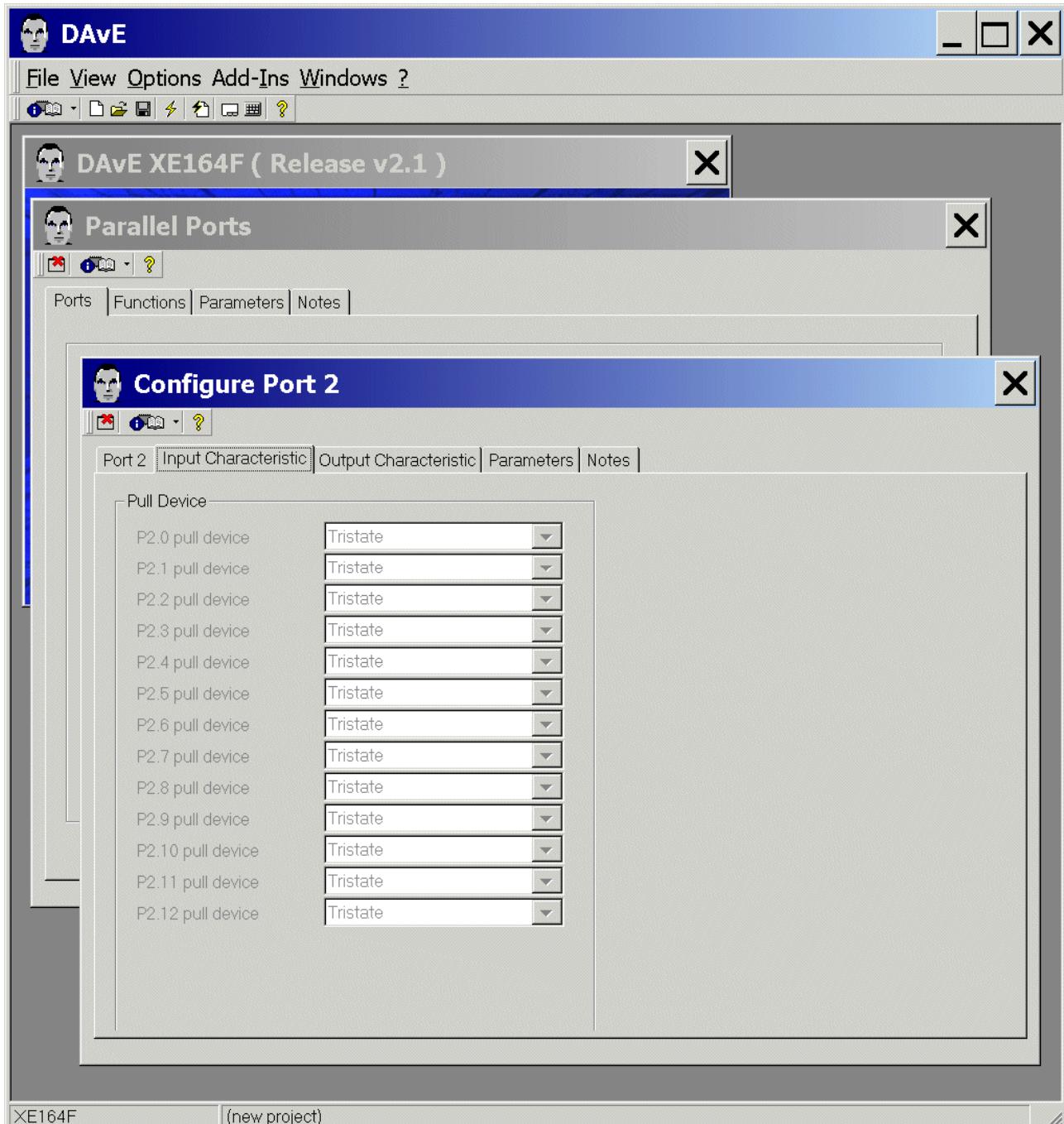
Ports: click “Configure Port 2”



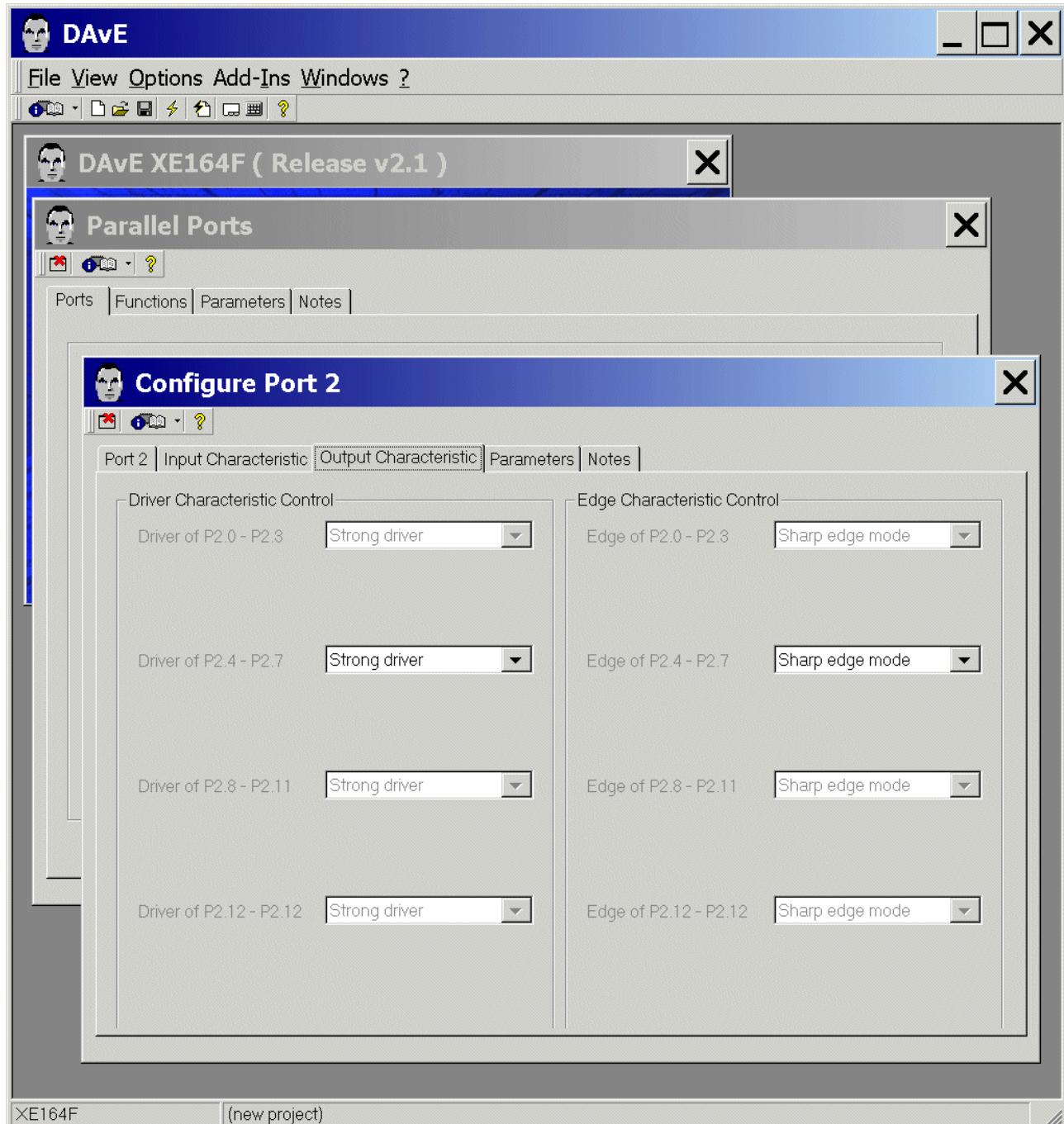
Port 2: Functionality: tick  Use P2.7 as general IO - Direction: click  Out



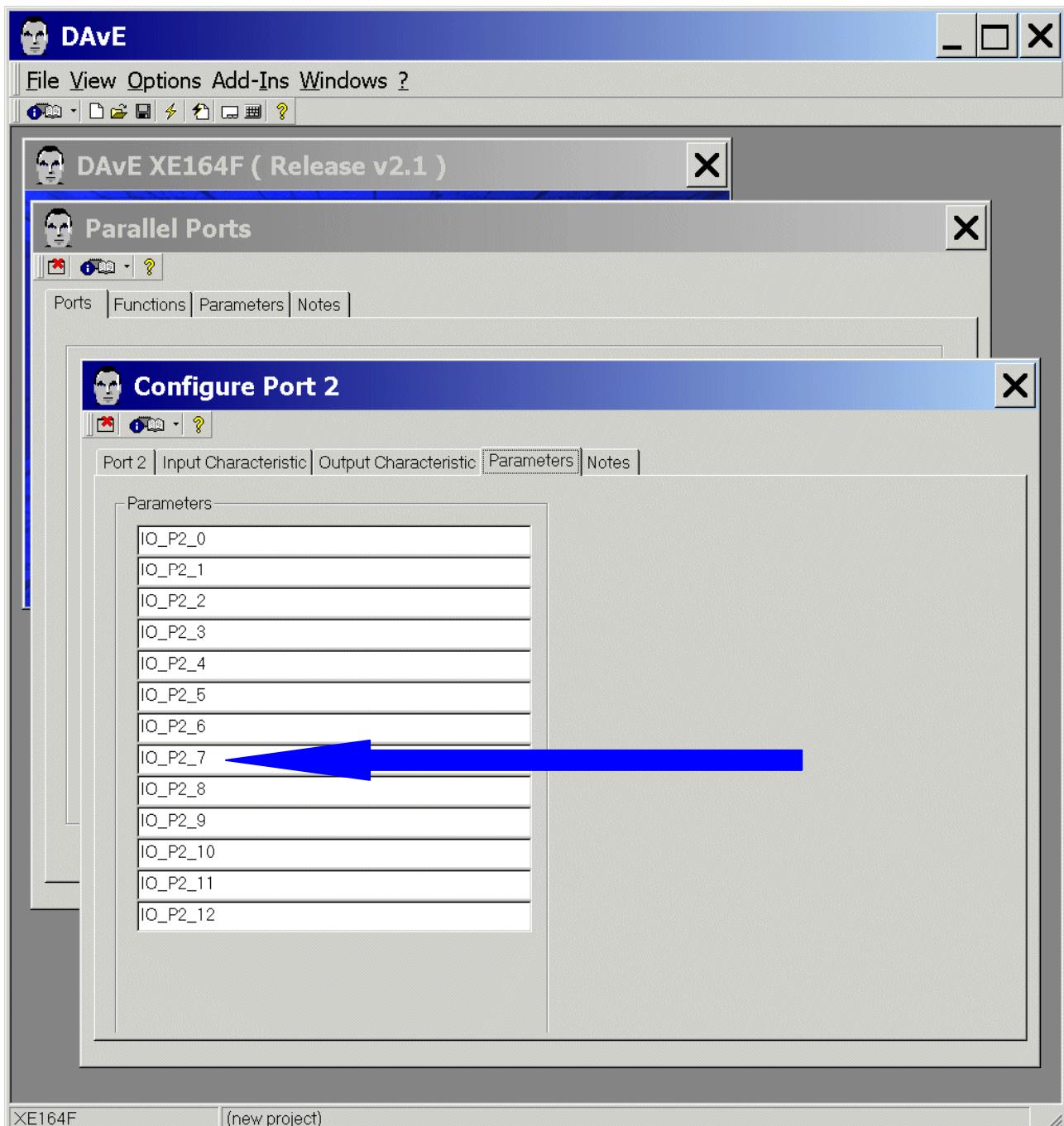
Input Characteristic: (do nothing)



**Output Characteristic:** (do nothing)



Parameters: (do nothing)

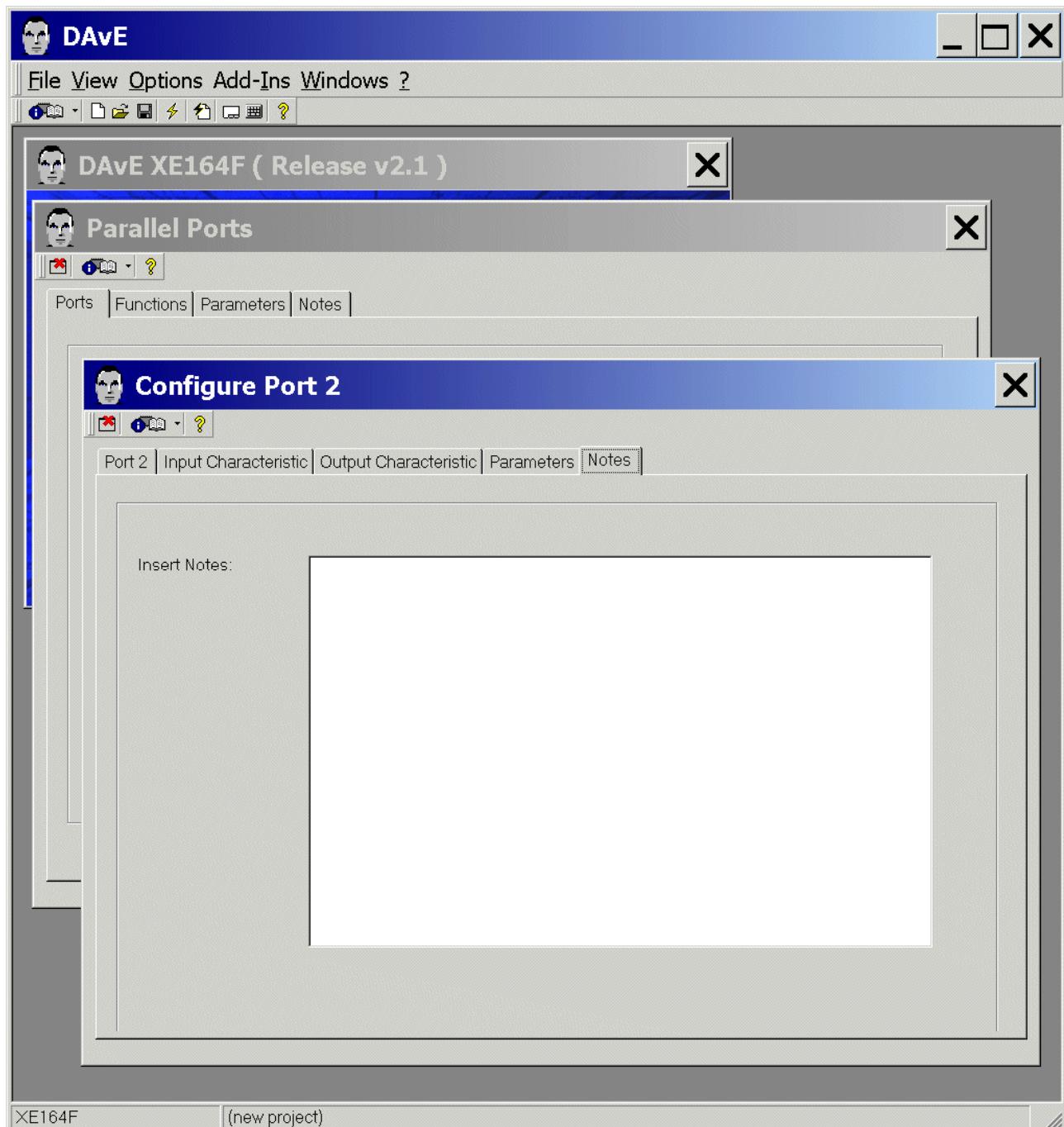


Note:

We will use the name **IO\_P2\_7** in application programming.

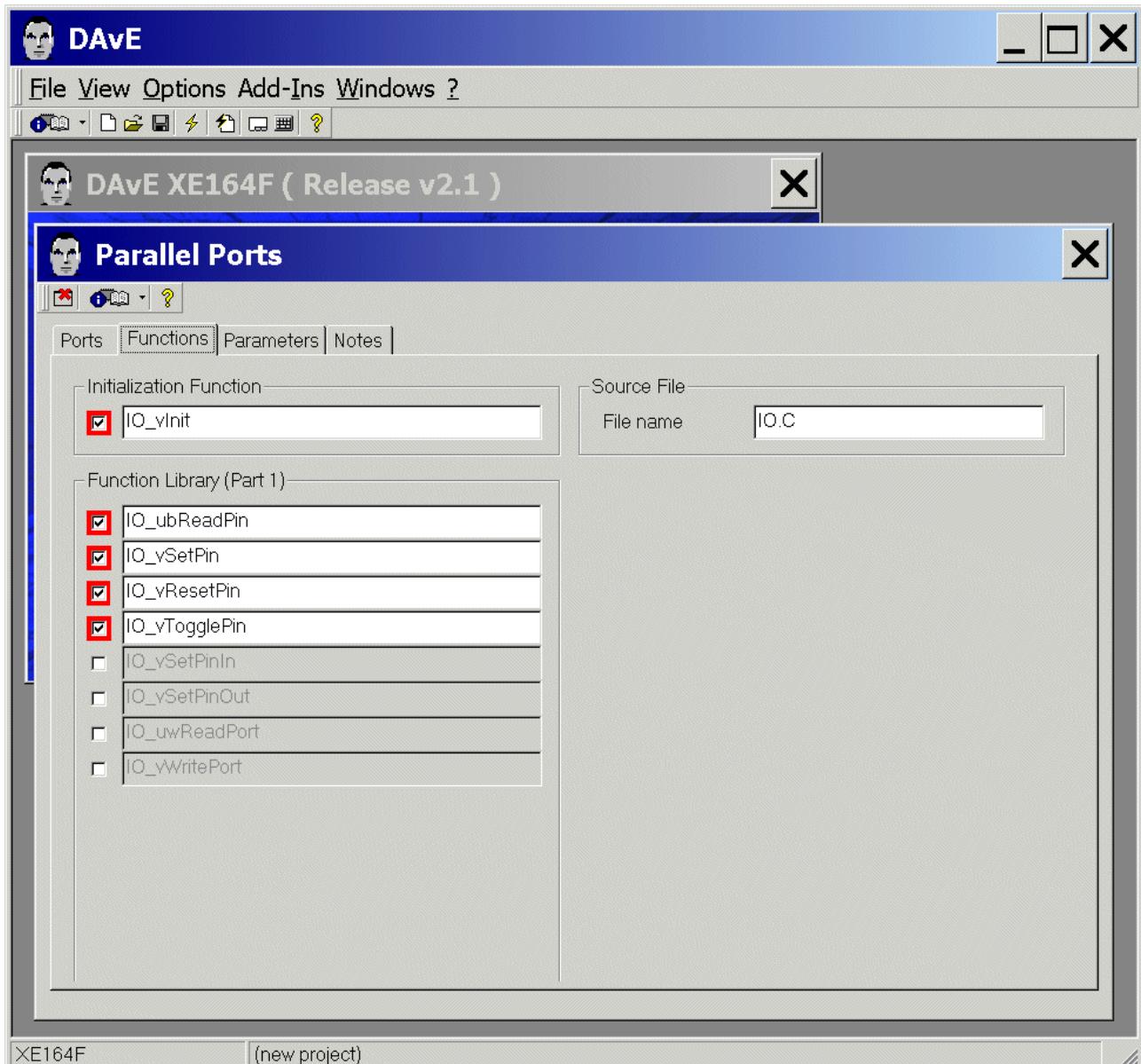


Notes: Insert Notes: If you wish, you can insert your comments here.

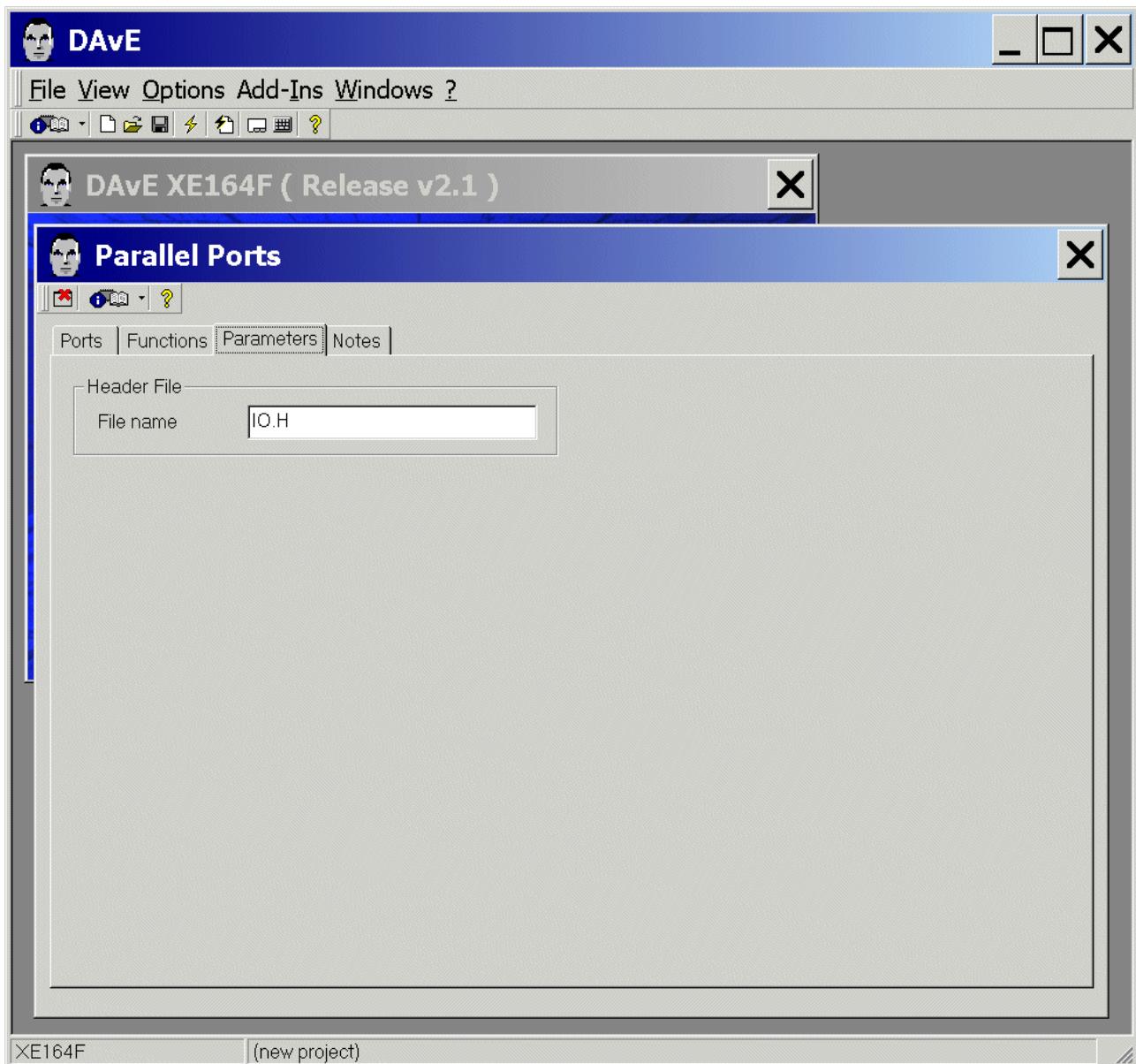


Exit and Save this dialog now by clicking  the close button:

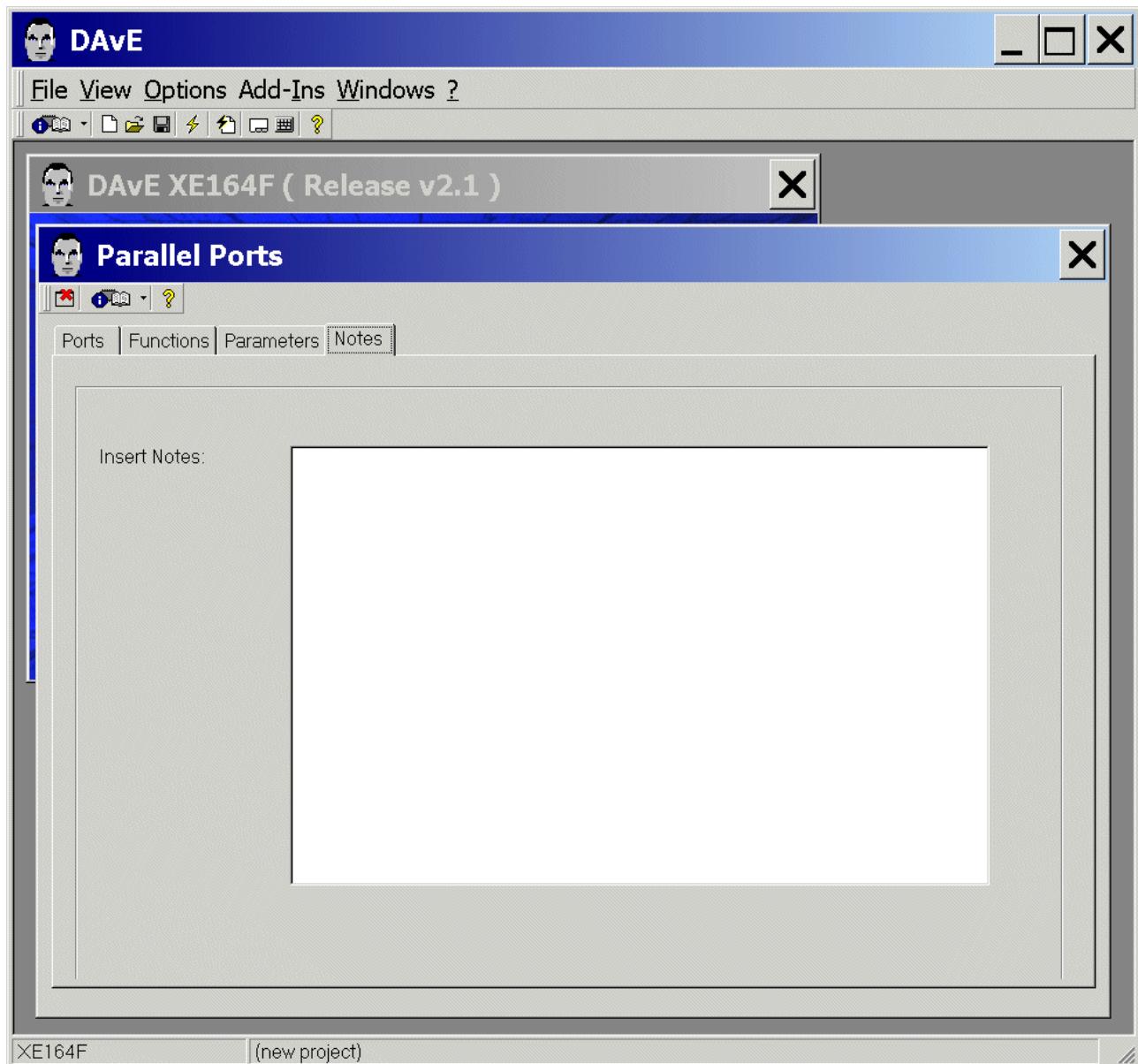
Functions: Initialization Functions: tick/check  IO\_vInit  
Functions: Function Library (Part 1): tick  IO\_ubReadPin  
Functions: Function Library (Part 1): tick  IO\_vSetPin  
Functions: Function Library (Part 1): tick  IO\_vResetPin  
Functions: Function Library (Part 1): tick  IO\_vTogglePin



Parameters: (do nothing)



Notes: Insert Notes: If you wish, you can insert your comments here.



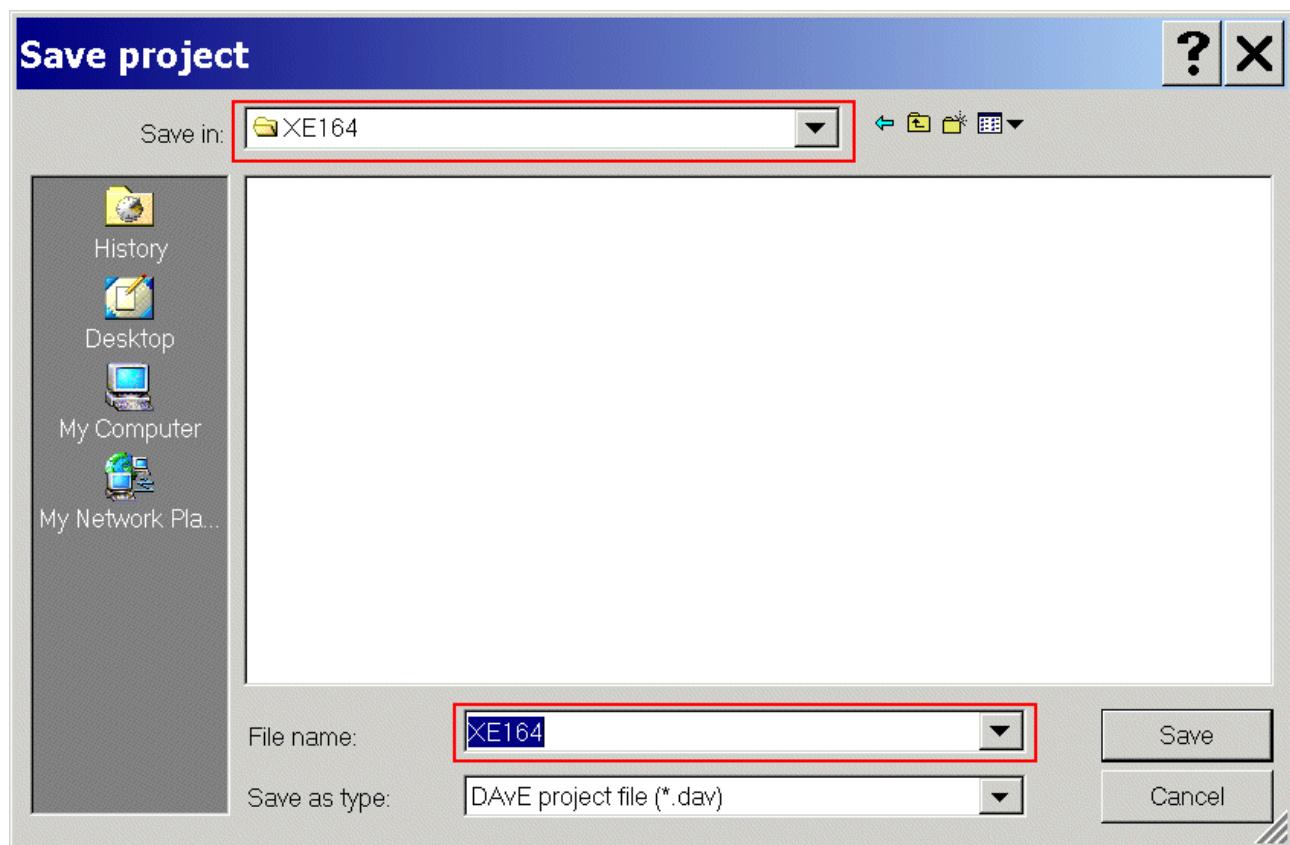
Exit and Save this dialog now by clicking  the close button.

Save the project:

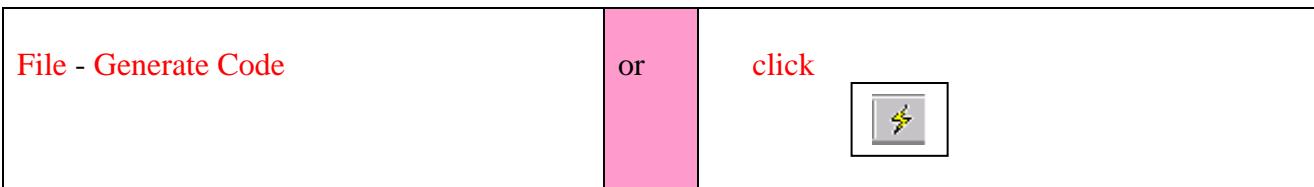
File  
Save



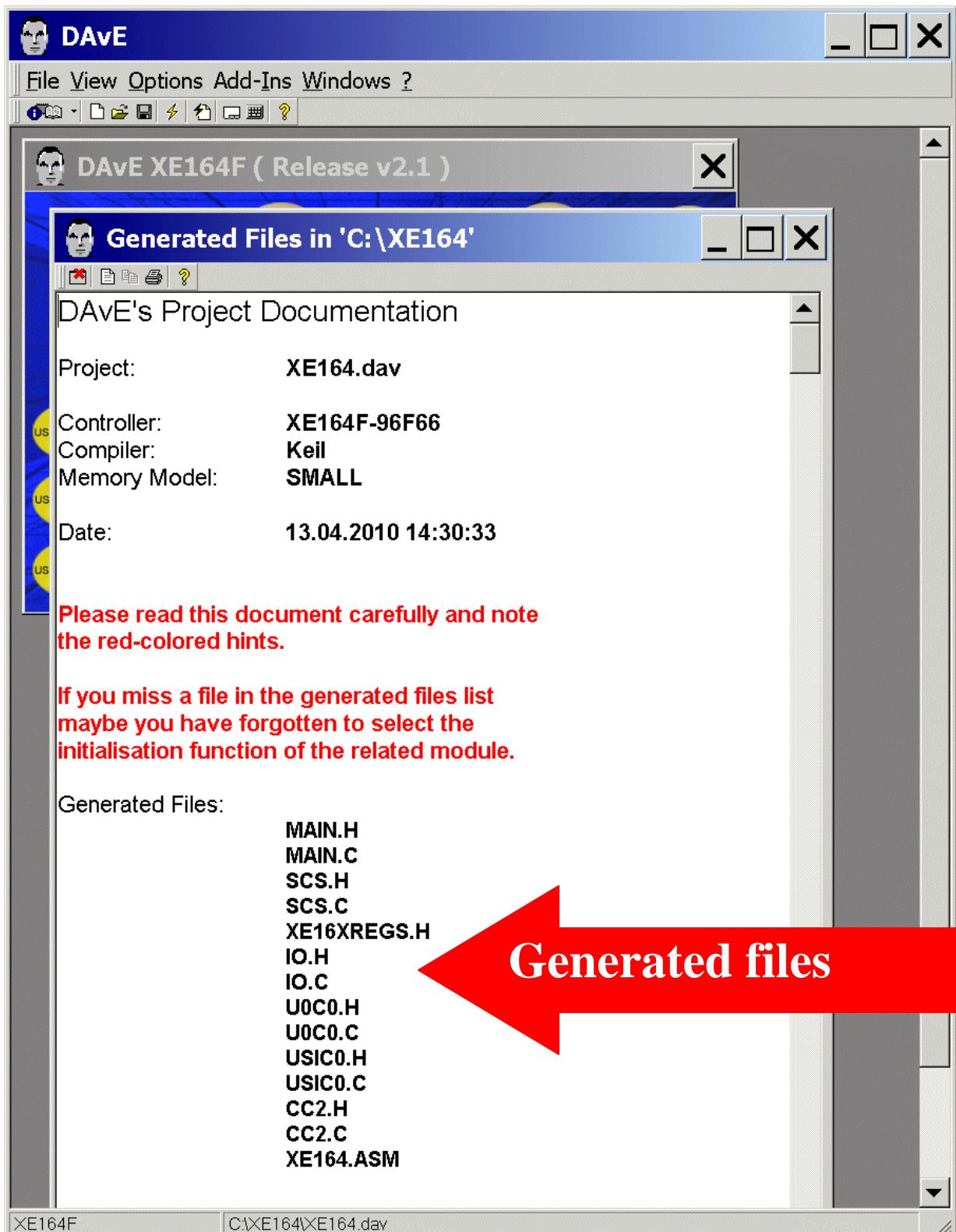
Save project: Save in C:\XE164 ( create new directory  
File name: XE164



Save

Generate Code:

DAvE will show you all the files he has generated  
(File Viewer opens automatically):

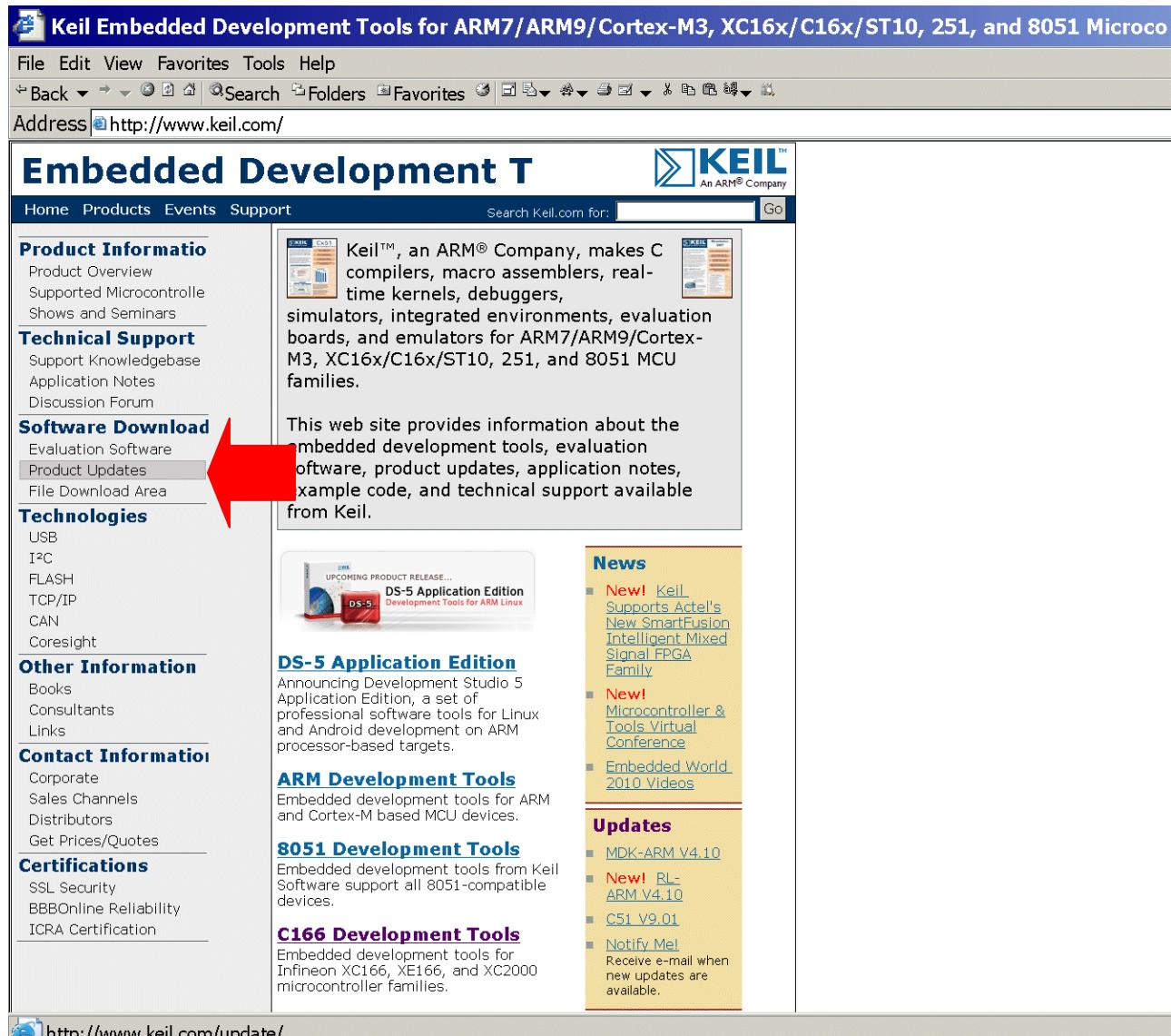


File - Exit  
 Save changes?

Click: Yes

#### 4.) Using the KEIL - μVision 4 Development Tools:

Download the tool chain: You can download the Keil Development Tools @ <http://www.keil.com> :



The screenshot shows the Keil Embedded Development Tools website. The left sidebar includes links for Product Information, Technical Support, Software Download (with 'Product Updates' highlighted), Technologies, Other Information, Contact Information, and Certifications. The main content area features a large text block about Keil's products, a 'DS-5 Application Edition' announcement, and sections for ARM Development Tools, 8051 Development Tools, and C166 Development Tools. A red arrow points from the 'Product Updates' link in the sidebar to the 'Product Updates' section in the main content area.

Click Product Updates

**Keil Product Update Information - Infineon Technologies**

File Edit View Favorites Tools Help

Back Search Folders Favorites

Address http://www.keil.com/update/ Go

# Product Updates

Home Products Events Support

Search Keil.com for: Go

## Technical Support

- Overview
- Search
- Contact
- Assistance Request
- Feedback

## Support Resources

- Support Knowledgebase
- Article Index
- Top 10 Articles
- Product Manuals
- Application Notes
- File Download Area
- Product Updates**
- Update List Server
- Discussion Forum
- Books

## Product Information

- Software & Hardware Prc
- Why Buy Tools From Kei
- Contact
- Ordering Instructions
- Product Information Request
- Pricing Information Request
- Product Brochures
- Newsletters

## Overview

The following product updates are available for download.

## ARM Development Tools

- [RealView Microcontroller Development Kit Product Updates](#)
- [RealView Real-Time Library Product Updates](#)
- [CARM Product Updates](#)
- [ARTX-ARM Product Updates](#)

## C16x/ST10/XC16x Development Tools

- [C166 Product Updates](#)
- [ARTX-166 Product Updates](#)

## Cx51 Development Tools

- [C251 Product Updates](#)
- [C51 Product Updates](#)
- [LPC Development Studio Product Updates](#)

Contact Site Map Press Privacy Copyright © 2010 Keil™, An ARM® Company. All rights reserved.

Internet

Click C166 Product Updates

**C166 Product Updates - Infineon Technologies**

File Edit View Favorites Tools Help

Back Search Folders Favorites

Address http://www.keil.com/update/c166.asp Go

## Product Updates

Home Products Events Support

Search Keil.com for: Go

**Technical Support**

- Overview
- Search
- Contact
- Assistance Request
- Feedback

**Support Resources**

- Support Knowledgebase
- Article Index
- Top 10 Articles
- Product Manuals
- Application Notes
- File Download Area
- Product Updates
- Update List Server
- Discussion Forum
- Books

**Product Information**

- Software & Hardware Prc
- Why Buy Tools From Kei
- Contact
- Ordering Instructions
- Product Information Request
- Pricing Information Request
- Product Brochures
- Newsletters

## C166 Product Updates

- [C166 Version 7.00](#)
- [C166 Version 6.18](#)
- [C166 Version 6.17](#)
- [C166 Version 6.14](#)
- [C166 Version 6.11](#)
- [C166 Version 6.10a](#)
- [C166 Version 6.10](#)
- [C166 Version 6.09](#)
- [C166 Version 6.08](#)
- [C166 Version 6.06](#)
- [C166 Version 6.04a](#)
- [C166 Version 6.02](#)
- [C166 Version 6.00a](#)
- [C166 Version 5.05](#)
- [C166 Version 5.04a](#)
- [C166 Version 5.03](#)
- [C166 Version 5.01](#)
- [C166 Version 5.00](#)
- [C166 Version 4.27](#)
- [C166 Version 4.26a](#)
- [C166 Version 4.25](#)
- [C166 Version 4.24](#)
- [C166 Version 4.12](#)

**Note**

- Updates only work with C166 Version 4 and later. They will not work with C166 Version 3 or earlier.
- Always make a complete backup copy of your current installation before you attempt to perform any update.

Contact Site Map Press Privacy Copyright © 2010 Keil™, An ARM® Company. All rights reserved.

Internet



Click [C166 Version 7.00](#)

Insert your PRODUCT SERIAL NUMBER (PSN):

**C166 Version 7.00 Product Update - Infineon Technologies**

File Edit View Favorites Tools Help

Back ▶ Search Folders Favorites

Address <http://www.keil.com/update/sw/c166/7.00> Go

## Product Updates

KEIL An ARM® Company

Home Products Events Support Search Keil.com for: Go

### Technical Support

- Overview
- Search
- Contact
- Assistance Request
- Feedback

### Support Resources

- Support Knowledgebase
- Article Index
- Top 10 Articles
- Product Manuals
- Application Notes
- File Download Area
- Product Updates
- Update List Server
- Discussion Forum
- Books

### Product Information

- Software & Hardware Prc
- Why Buy Tools From Kei
- Contact
- Ordering Instructions
- Product Information Reque:
- Pricing Information Request
- Product Brochures
- Newsletters

## C166 Version 7.00 Product Update

The C166 Version 7.00 product update replaces components of your current Tool Set with the latest versions. It includes an update to μVision Version 4.00.

- See [What's New](#) in C166 Version 7.00.
- View the [Release Notes](#) for C166 Version 7.00.
- View the [Release Notes](#) for μVision Version 4.00.

### Downloading the Update

Complete the following form to download C166 Version 7.00.

**Enter Your Contact Information Below**  
(bold fields are required)

**LIC or PSN:** BUY82-98E87-KE9HD

**First Name:** Wilhelm

**Last Name:** Brezovits

**Professional Title:** FAE Microcontrollers

**E-mail:** wilhelm.brezovits@infineon.com

**Company:** Infineon Technologies Austria AG

**Company Web Site:** <http://www.infineon.com/microcontrollers>

**Address:** Linke Wienzeile 4/1/3

**City:** Wien

**State/Province:** Select Your State or Province

**Zip/Postal Code:** A-1060

**Country:** Austria

**Phone:** +43-676-4912589

**Fax:**

Send me e-mail when there is a new update.

**NOTICE:**  
If you select this check box, you **will** receive an e-mail message from Keil whenever a new update is available. If you don't wish to receive an e-mail notification, don't check this box.

PSN

Check our [Privacy Statement](#) for details regarding information collected on this web site.

Contact Site Map Press Privacy Copyright © 2010 Keil™, An ARM® Company. All rights reserved.

  Internet

Click

**C166 Version 7.00 Product Update - Infineon Technologies**

File Edit View Favorites Tools Help

Back Search Folders Favorites

Address <http://www.keil.com/update/sw/c166/7.00>

## Product Updates

Home Products Events Support

Search Keil.com for: Go

### Technical Support

- Overview
- Search
- Contact
- Assistance Request
- Feedback

### Support Resources

- Support Knowledgebase
- Article Index
- Top 10 Articles
- Product Manuals
- Application Notes
- File Download Area
- Product Updates
- Update List Server
- Discussion Forum
- Books

### Product Information

- Software & Hardware Prc
- Why Buy Tools From Kei
- Contact
- Ordering Instructions
- Product Information Reque:
- Pricing Information Request
- Product Brochures
- Newsletters

## C166 Version 7.00 Product Update

- Right-click on **C166V700.EXE** and save it to your computer.
- PDF files may be opened with Acrobat Reader.
- ZIP files may be opened with PKZIP or WINZIP.

**C166V700.EXE** (25,823K)  
Monday, October 26, 2009

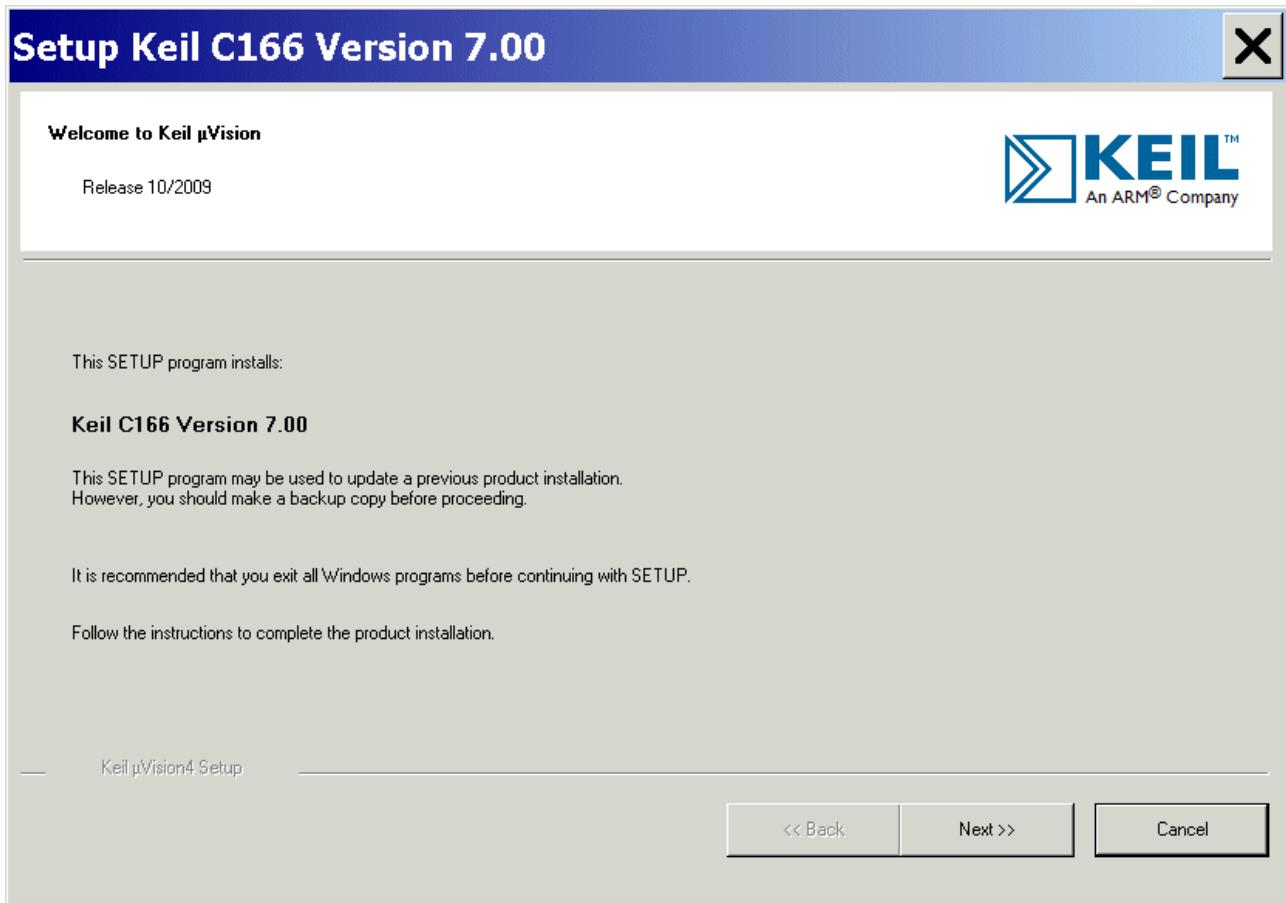
Estimated File Download Time:  
< 1.9 Hours: 56Kb Modem  
< 50 Minutes: 128Kb ISDN  
< 4 Minutes: T1/Broadband



Contact Site Map Press Privacy Copyright © 2010 Keil™, An ARM® Company. All rights reserved.

Internet

Download and Execute  C166V700.EXE ( - or any higher version )  
and install the Keil tool chain.



## Setup Keil C166 Version 7.00

### License Agreement

Please read the following license agreement carefully.



To continue with SETUP, you must accept the terms of the License Agreement. To accept the agreement, click the check box below.

### End-User License Agreement for ARM Keil Software Development Tools

THIS END USER LICENCE AGREEMENT ("LICENCE") IS A LEGAL AGREEMENT BETWEEN YOU (EITHER A SINGLE INDIVIDUAL, OR SINGLE LEGAL ENTITY) AND ARM LIMITED ("ARM") FOR THE USE OF THE SOFTWARE ACCOMPANYING THIS LICENCE. ARM IS ONLY WILLING TO LICENSE THE SOFTWARE TO YOU ON CONDITION THAT YOU ACCEPT ALL OF THE TERMS IN THIS LICENCE. BY CLICKING "I AGREE" OR BY INSTALLING OR OTHERWISE USING OR COPYING THE SOFTWARE YOU INDICATE THAT YOU AGREE TO BE BOUND BY ALL THE TERMS OF THIS LICENCE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENCE, DO NOT USE THE SOFTWARE.

I agree to all the terms of the preceding License Agreement

Keil µVision4 Setup

<< Back

Next >>

Cancel

## Setup Keil C166 Version 7.00

### Folder Selection

Select the folder where SETUP will install files.



SETUP will install µVision4 in the following folder.

To install to this folder, press 'Next'. To install to a different folder, press 'Browse' and select another folder.

Destination Folder

C:\Program Files\C166\_v7.0.0

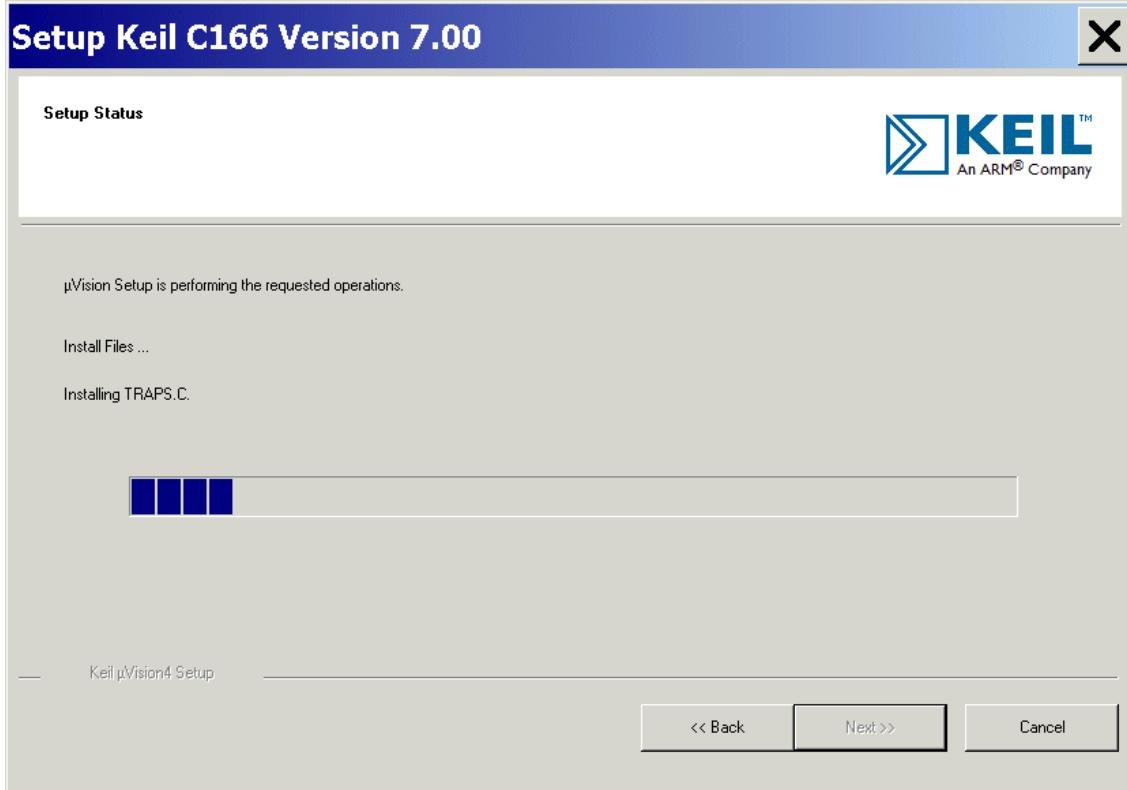
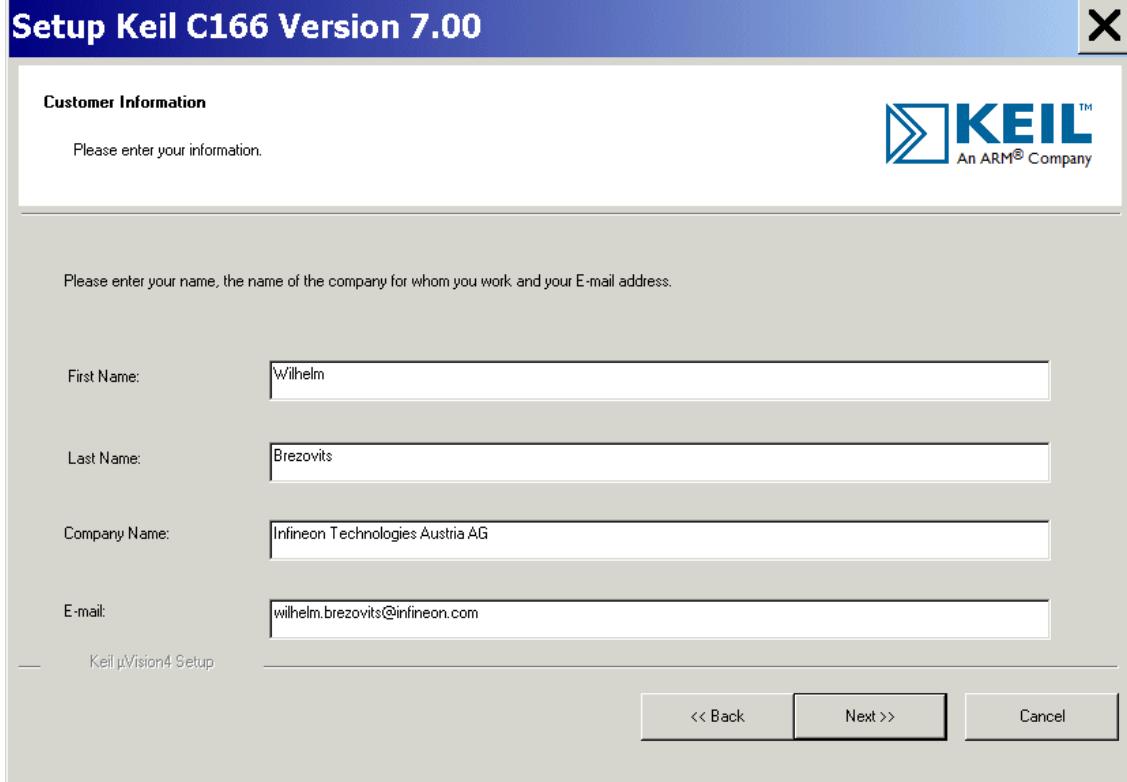
Browse ...

Keil µVision4 Setup

<< Back

Next >>

Cancel



## Setup Keil C166 Version 7.00



Keil µVision4 Setup completed



µVision Setup has performed all requested operations successfully.

Show Release Notes.

Retain current µVision configuration.

Add example projects to the recently used project list.

Keil µVision4 Setup

<< Back

Finish

Cancel

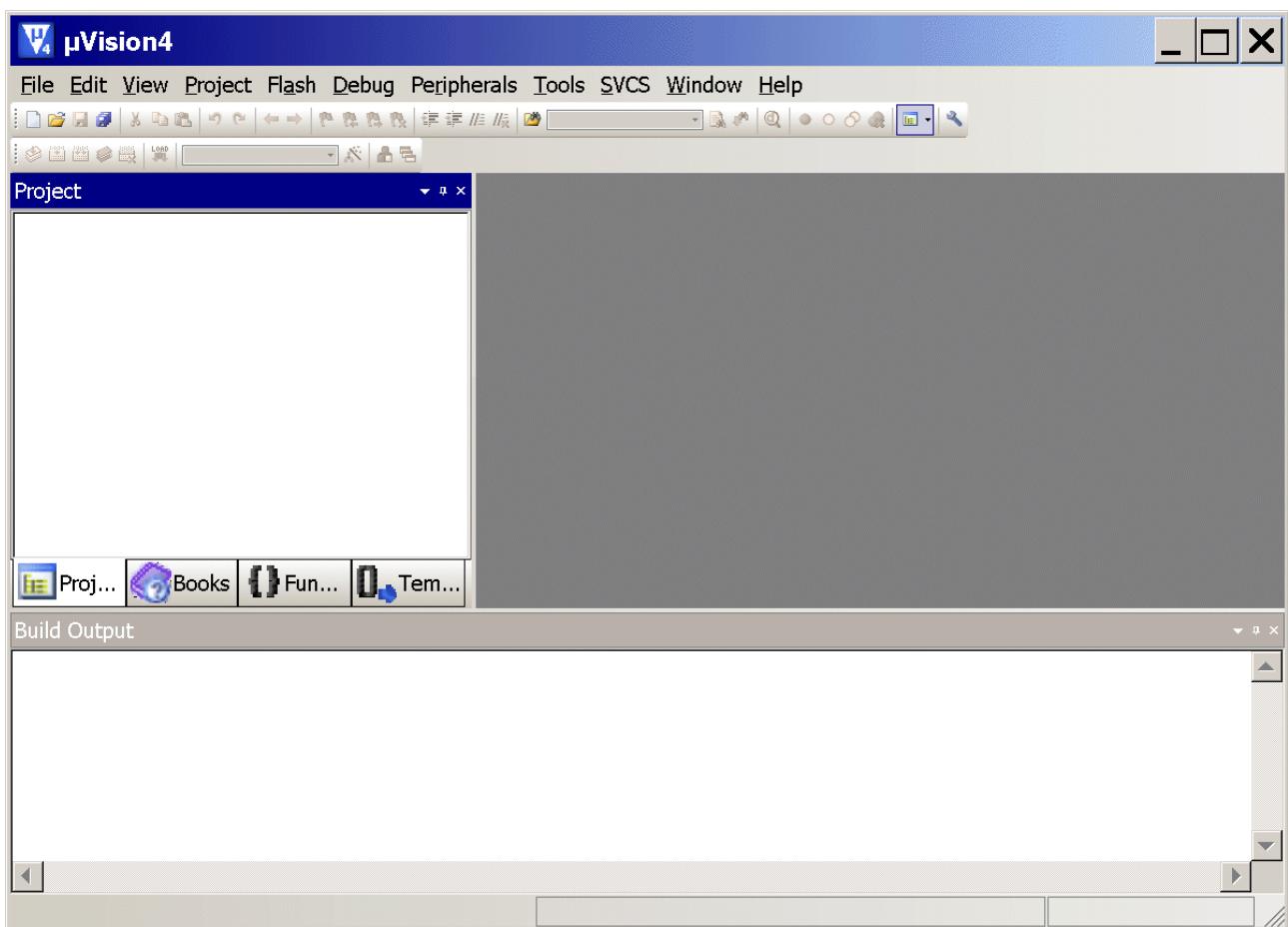
Finish

Click



Start Keil μVision4 and open the DAvE Project:

If you see an open project – close it: **Project - Close Project**

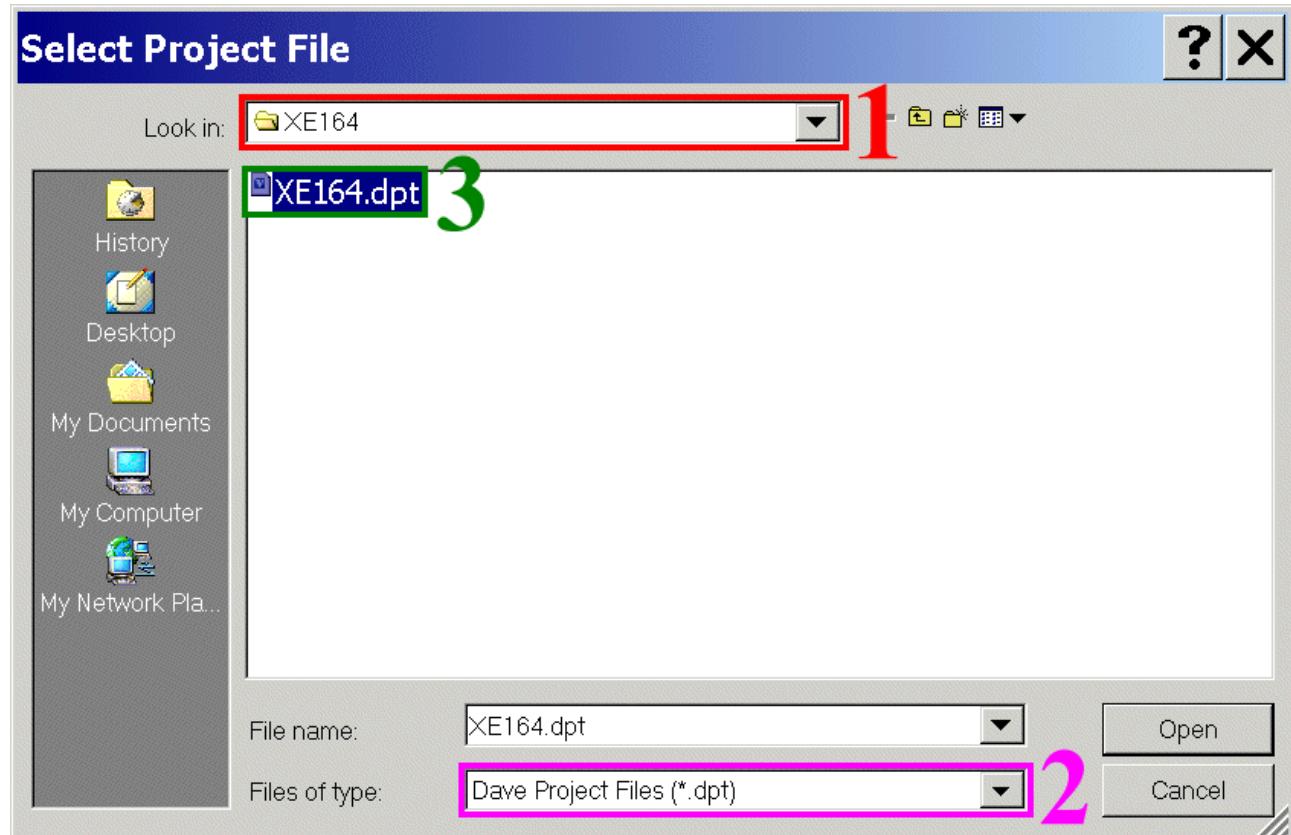


### Project - Open Project

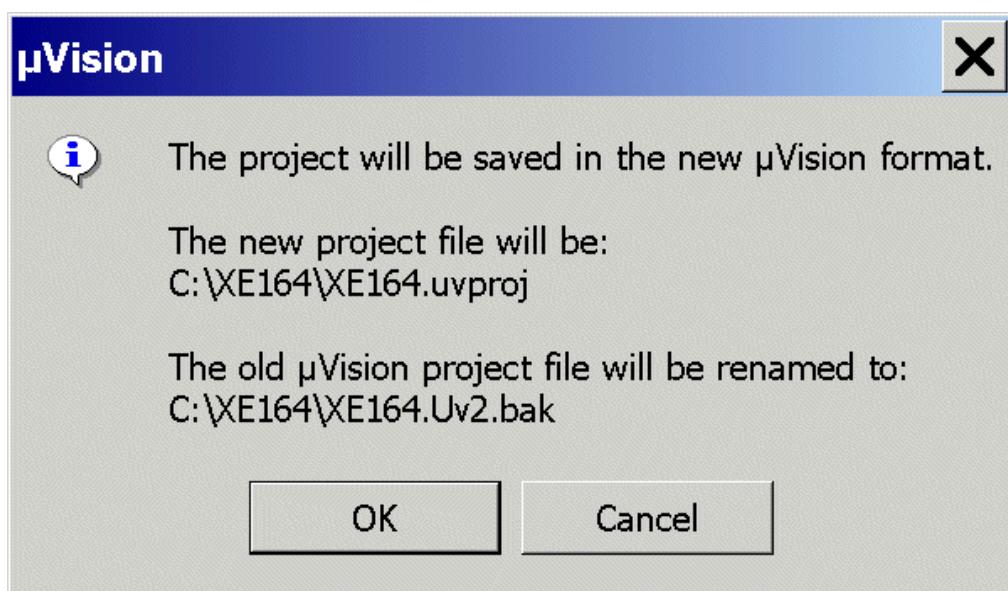
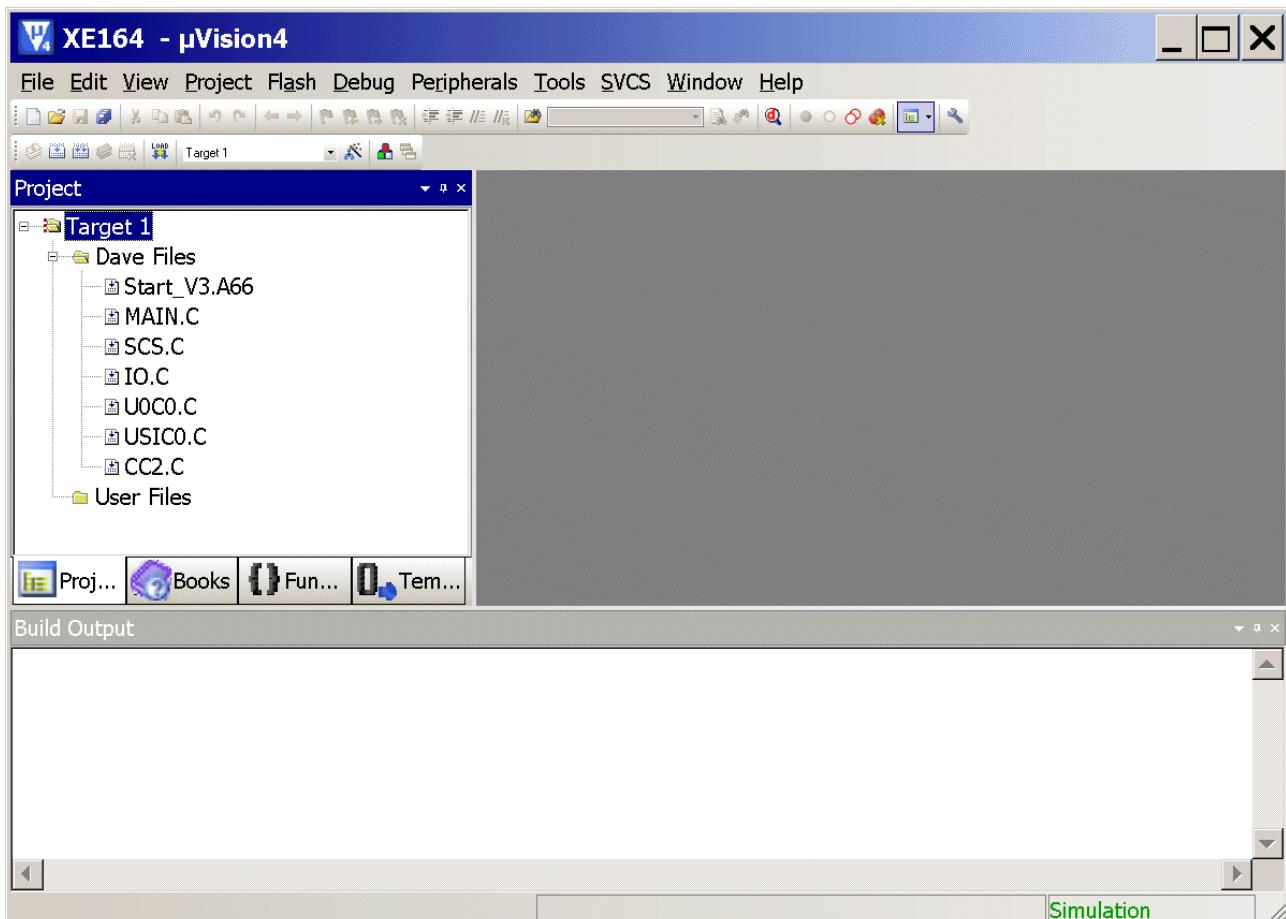
Select Project File: Look in: choose C:\XE164 (1)

Select Project File: Files of type: select Dave Project Files (2)

Click XE164.dpt (3)

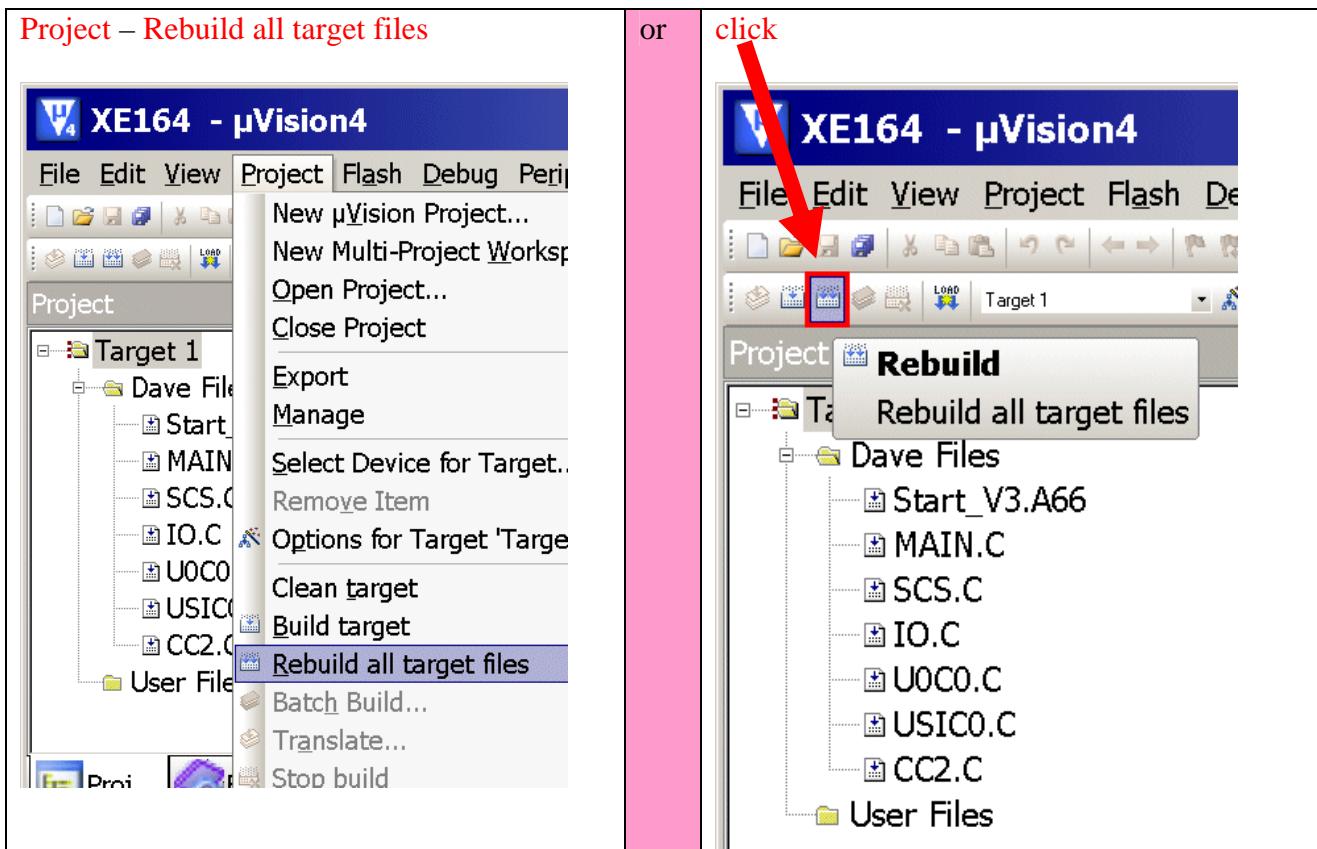


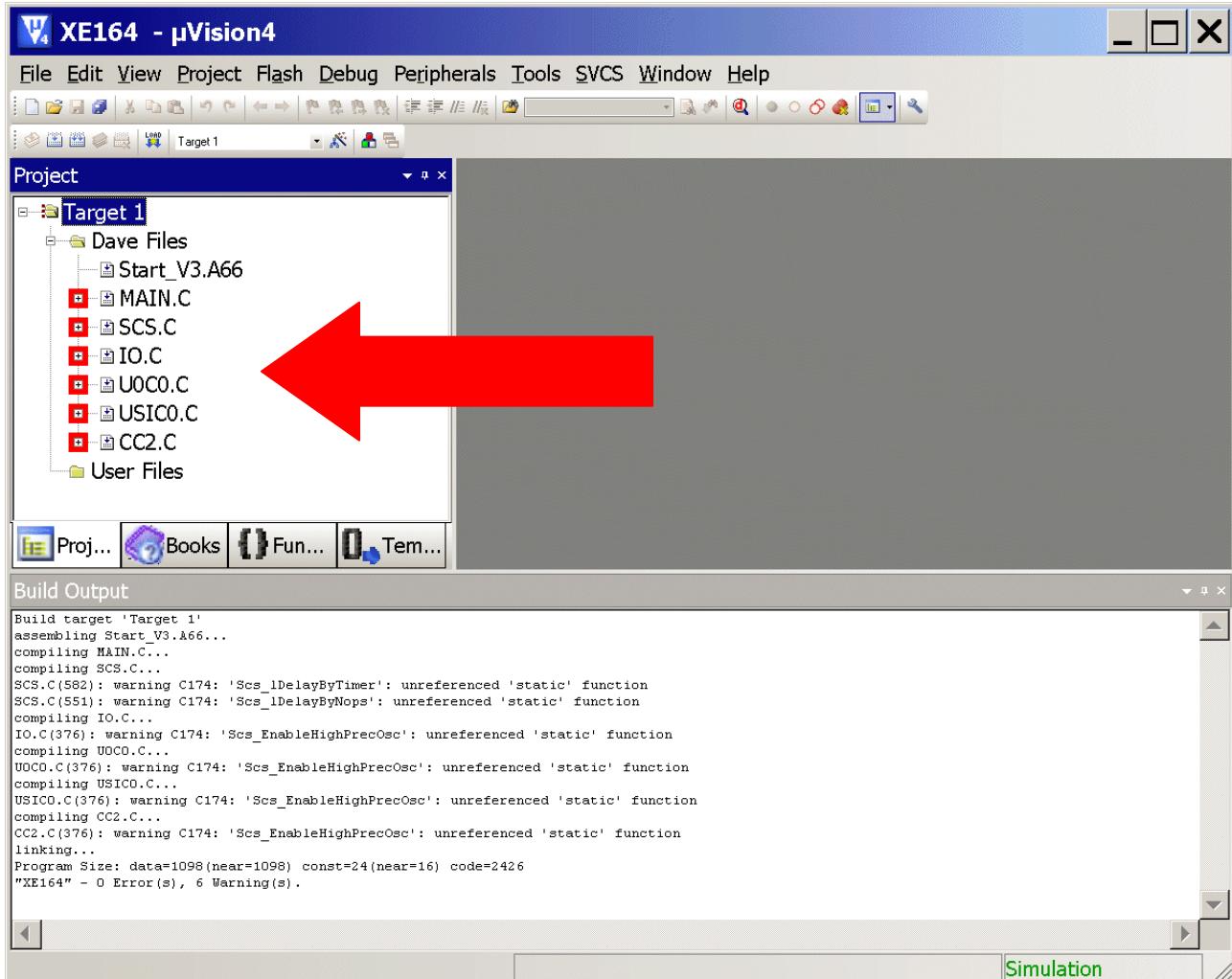
Click Open



( Click **OK** )

Generate „make“- file:





### Note:

This step generates a makefile and shows the include files.

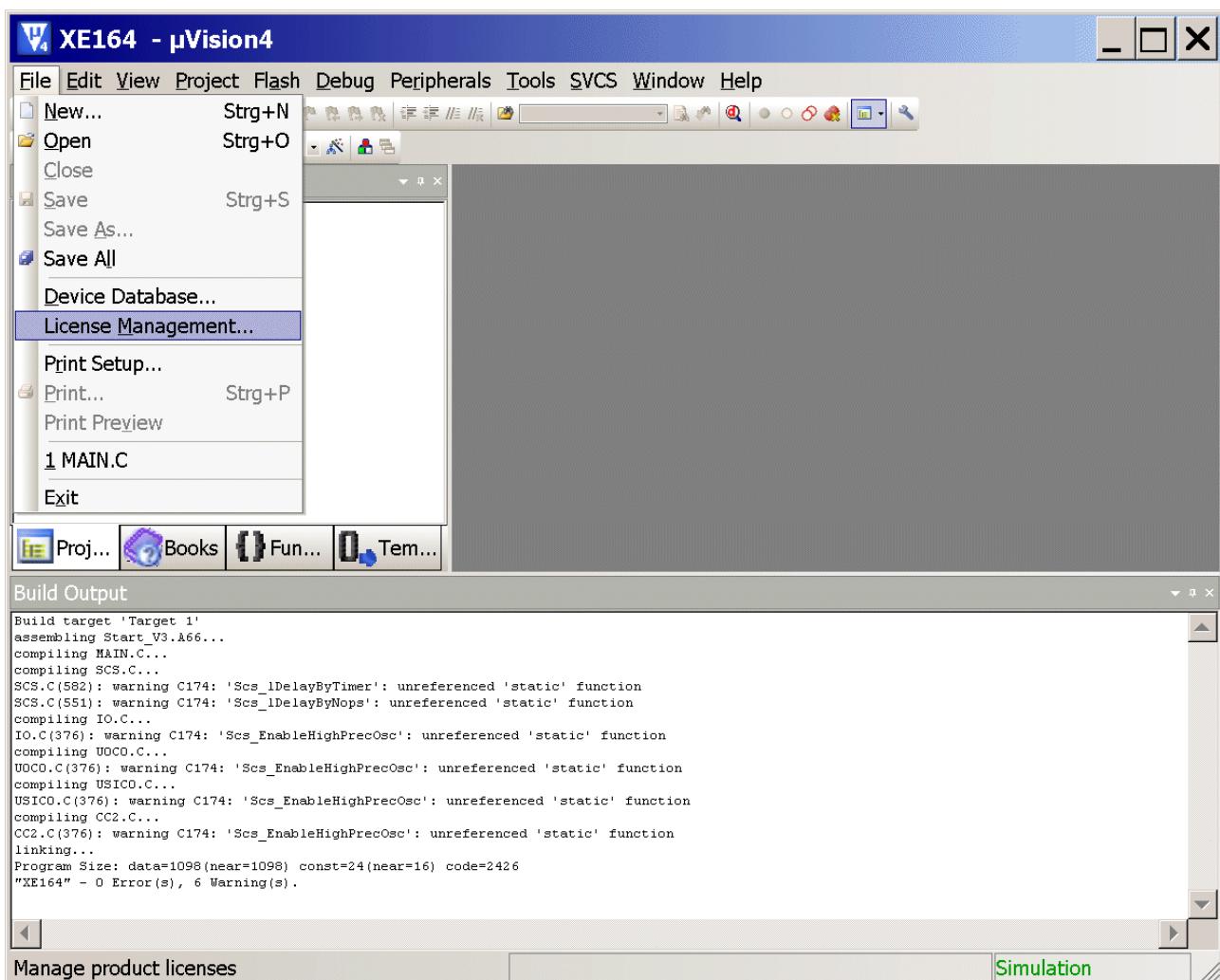


## uVision4 License Management

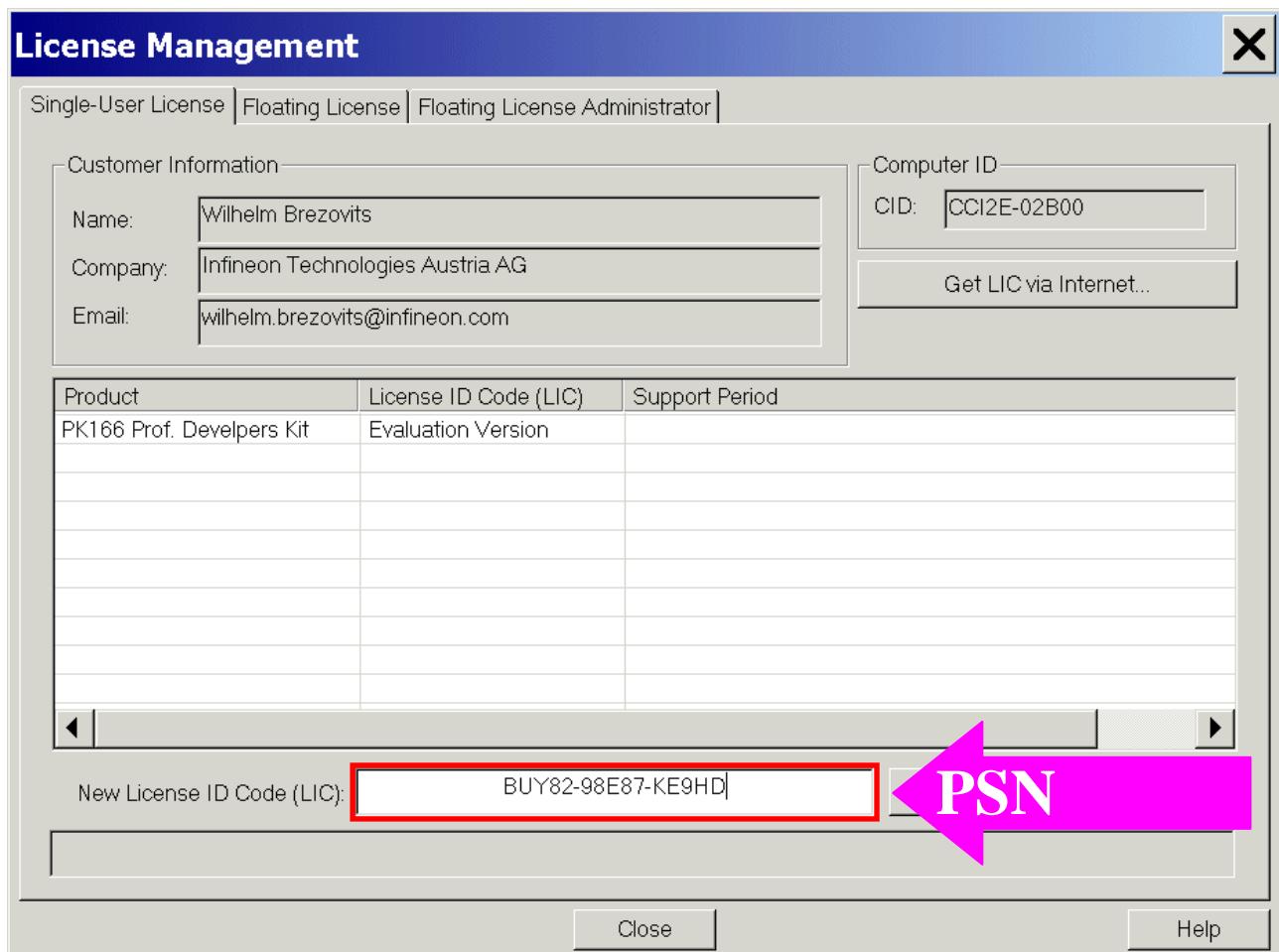
**Note:**

To activate your Keil product, copy and paste your PRODUCT SERIAL NUMBER (PSN) into the New License ID Code (LIC) input field in the uVision4 License Manager Dialog (available from the File menu).

### File - License Management...



**License Management:** Single-User License: New License ID Code (LIC):  
Copy and paste your PRODUCT SERIAL NUMBER (PSN) into the New License ID Code (LIC)  
input field in the uVision4 License Manager Dialog.



Click

Get LIC via Internet...

## Obtaining a License ID Code (LIC)



You are about to register a Product and obtain a License ID Code (LIC).

The following information must be sent to [www.keil.com](http://www.keil.com):

Product Serial Number (PSN)

Computer ID Number (CID): CCI2E-02B00

Notes:

- You may register the product manually at <http://www.keil.com/license/install.htm>.
- After registering, you will receive an e-mail containing the LIC for your computer.
- Enter the LIC into the New License ID Code field.

OK

Cancel

OK

Click

Insert your PRODUCT SERIAL NUMBER (PSN) and contact information:

**Single-User License Management - Infineon Technologies**

File Edit View Favorites Tools Help

Back Search Folders Favorites

Address http://www.keil.com/license/install.htm?C=CCI2E-02B00&E=wilhelm.brezovits@infineon.com&F=Wil

## Product Licensing

Home Products Events Support

Search Kell.com for: Go

**Product Licensing**

- Overview
- Single-User License
- Single-User License Unir
- Floating-User License

### Single-User License

Enter your Product Serial Number (PSN) and contact information using the following form to license your Keil product.

**Please make certain your e-mail address is valid.** After verifying your Product Serial Number and Computer ID (CID), **we will send you a License ID Code (LIC) via e-mail.** E-mail is sent from licmgr@keil.com so make sure any spam blocker you use is configured to allow this address.

**Enter Your Contact Information Below**  
(bold fields are required)

<b>Computer ID (CID):</b> CCI2E-02B00	<b>Product Serial # (PSN):</b> BUY82-98E87-KE9HD
<b>PC Description:</b> Notebook Enter a description of the PC on which this license is registered. For example: LAB PC, Office Computer, Laptop, John's PC, etc.	
<b>First Name:</b> Wilhelm	<b>Last Name:</b> Brezovits
<b>Professional Title:</b> FAE Microcontrollers	
<b>E-mail:</b> wilhelm.brezovits@infineon.com	<b>Company:</b> Infineon Technologies Austria AG
<b>Company Web Site:</b> http://www.infineon.com/microcontrollers	
<b>Address:</b> Linke Wienzeile 4/1/3	
<b>City:</b> Wien	

**Send me e-mail when there is a new update.**  
**NOTICE:**  
If you select this check box, you **will** receive an e-mail message from Keil whenever a new update is available. If you don't wish to receive an e-mail notification, don't check this box.

**Submit** **Reset**

Check our [Privacy Statement](#) for details regarding information collected on this web site.

Contact Site Map Press Privacy Copyright © 2010 Keil™, An ARM® Company. All rights reserved.

Internet

Click **Submit**

**Single-User License Management - Infineon Technologies**

File Edit View Favorites Tools Help

Back ▾ Forward ▾ Stop ▾ Home ▾ Search ▾ Folders ▾ Favorites ▾

Address: http://www.keil.com/license/install.htm

Go

## Product Licensing

KEIL™ An ARM® Company

Home Products Events Support

Search Keil.com for:  Go

### Product Licensing

- Overview
- Single-User License
- Single-User License Unit
- Floating-User License

### Thanks for Licensing Your Product

We have sent your product registration information including the License ID Code (LIC) via e-mail to **wilhelm.brezovits@infineon.com**.

When you receive this e-mail, copy the License ID Code (LIC) and paste it into the **New License ID Code** input field in the µVision License Manager Dialog — Single-User License Tab (available from the File Menu).

If you have multiple Keil products you may [Register Another Product](#) at this time.

#### Quick Links

- Licensing User's Guide

#### Knowledgebase

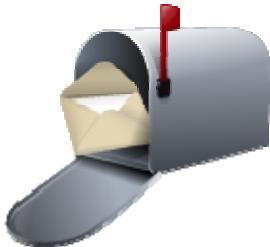
- RE-LICENSING ISSUES
- UNABLE TO REGISTER PRODUCT SERIAL NUMBER (PSN)
- LICENSING A PC NOT CONNECTED TO THE INTERNET

Top 3 License Issues

Contact Site Map Press Privacy Copyright © 2010 Keil™, An ARM® Company. All rights reserved.

Error on page. Internet

You've got mail!



Check your Email:

**Keil Product License - Message (Plain Text)**

**File Edit View Insert Format Tools Actions SECURE secure mail Help**

**Reply | Reply to All | Forward |**

From: licmgr@keil.com Sent: Mi 14.04.2010 14:16  
To: Brezovits Wilhelm (IFAT SMD IMM FAE)  
Cc:  
Subject: Keil Product License

Thank you for licensing your Keil product. Your License ID Code (LIC) is printed below. Print a copy of this e-mail to save with your installation CD.

PK166 Professional Developer's Kit  
Support Ends 31 May 2011

PC Description : NOTEBOOK  
Computer ID (CID): CCI2E-02B00  
Serial Number (PSN): BUY82-98E87-KE9HD

License ID Code (LIC): UJ01B-2DJXL-6T93S-JM6G8-21WVS-ZEPYZ

To activate your Keil product, copy the License ID Code (LIC) and paste it into the New License ID Code input field on the Single-User License Tab in the uVision3 License Manager Dialog (available from the File menu).

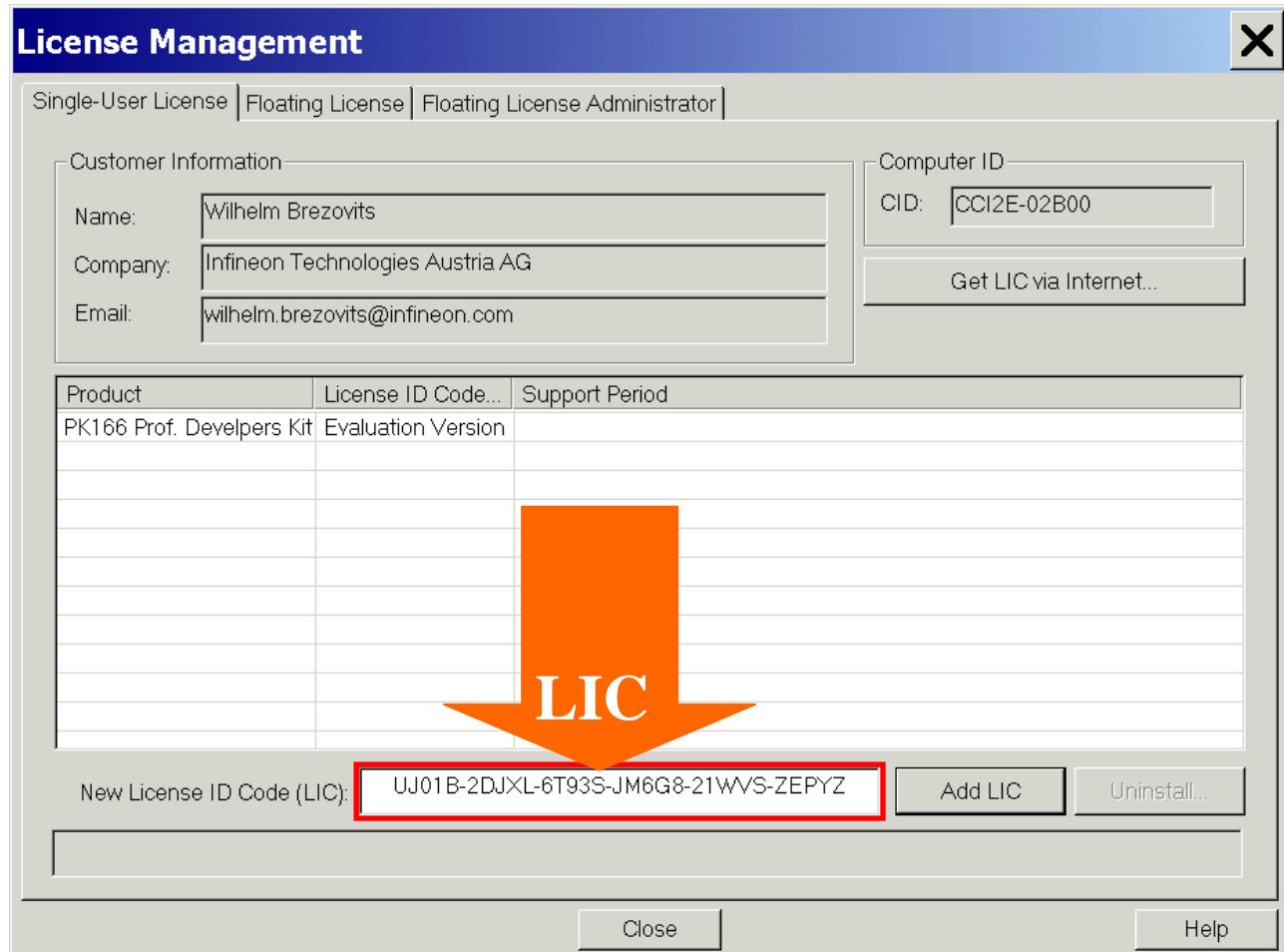
\*\*\* DO NOT REPLY TO THIS EMAIL: For licensing problems or questions, please contact Keil Technical Support.

Thank You,  
Technical Support

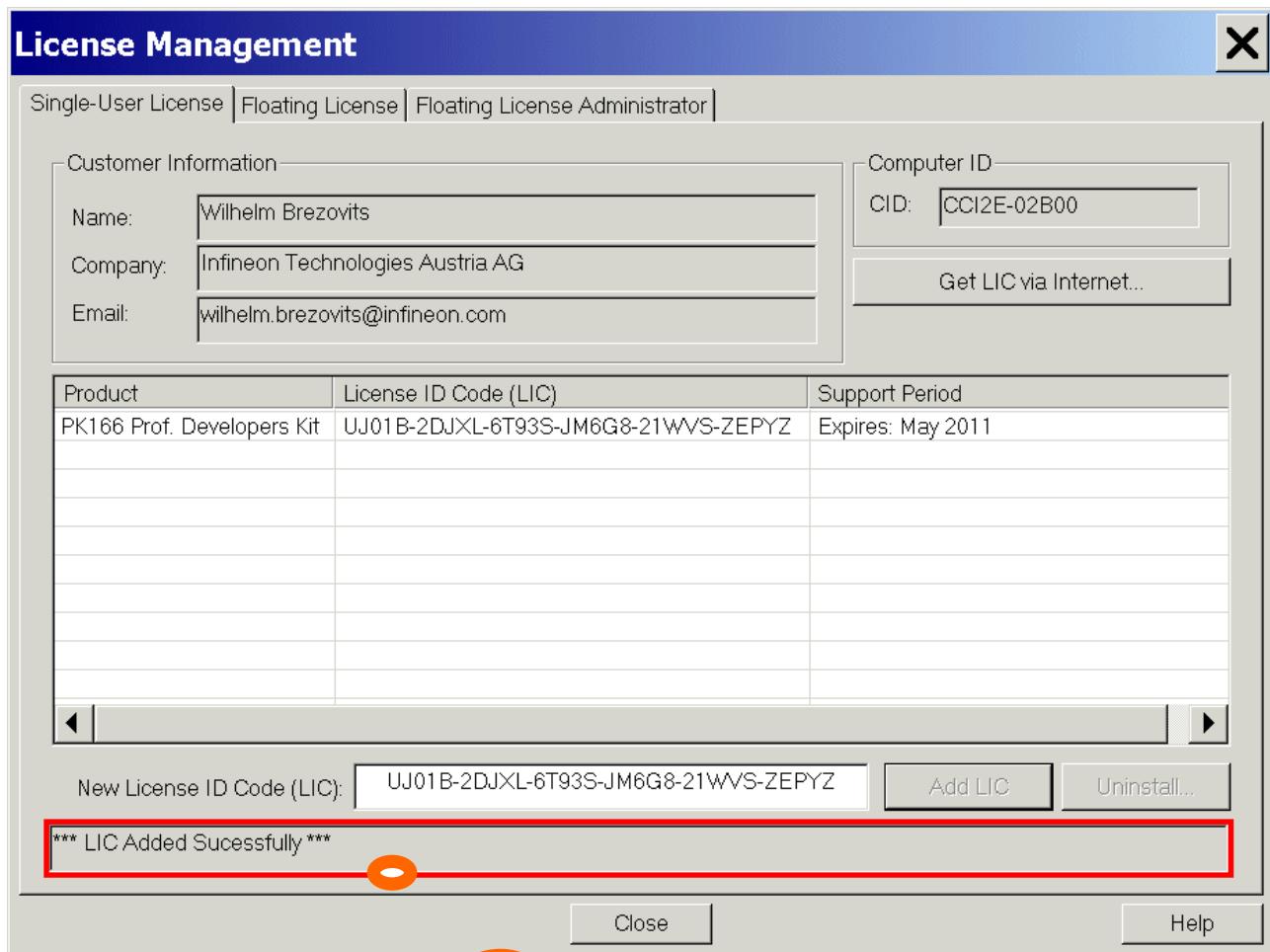
Keil, An Arm Company  
1501 10th Street, Suite 110  
Plano, Texas 75074 USA  
Phone: +1 972-312-1107  
FAX: +1 972-312-1159  
<http://www.keil.com/>

IMPORTANT NOTICE: The contents of this email and any attachments are confidential and may also be privileged. If you are not the named recipient, please notify the sender immediately and do not disclose the contents to any other person, use it for any purpose, or store or copy the information in any medium.

Insert your License ID Code (LIC):



Click **Add LIC**



**Close**



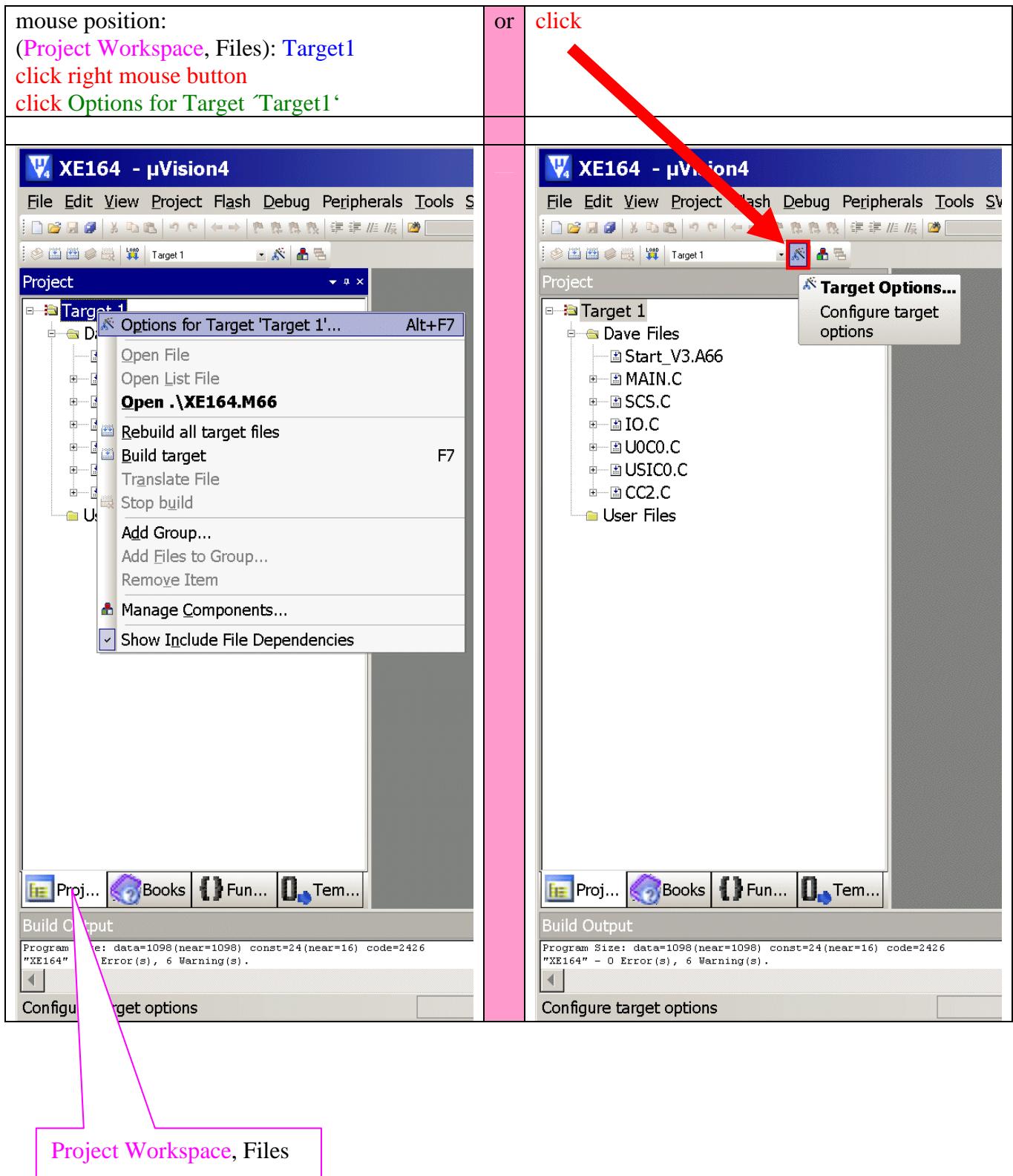
### Note:

Now you can write software for Infineon's 16 bit microcontrollers without the limitations (4 KBytes) of the Evaluation Compiler. That means your program may be as large as there is program memory available.

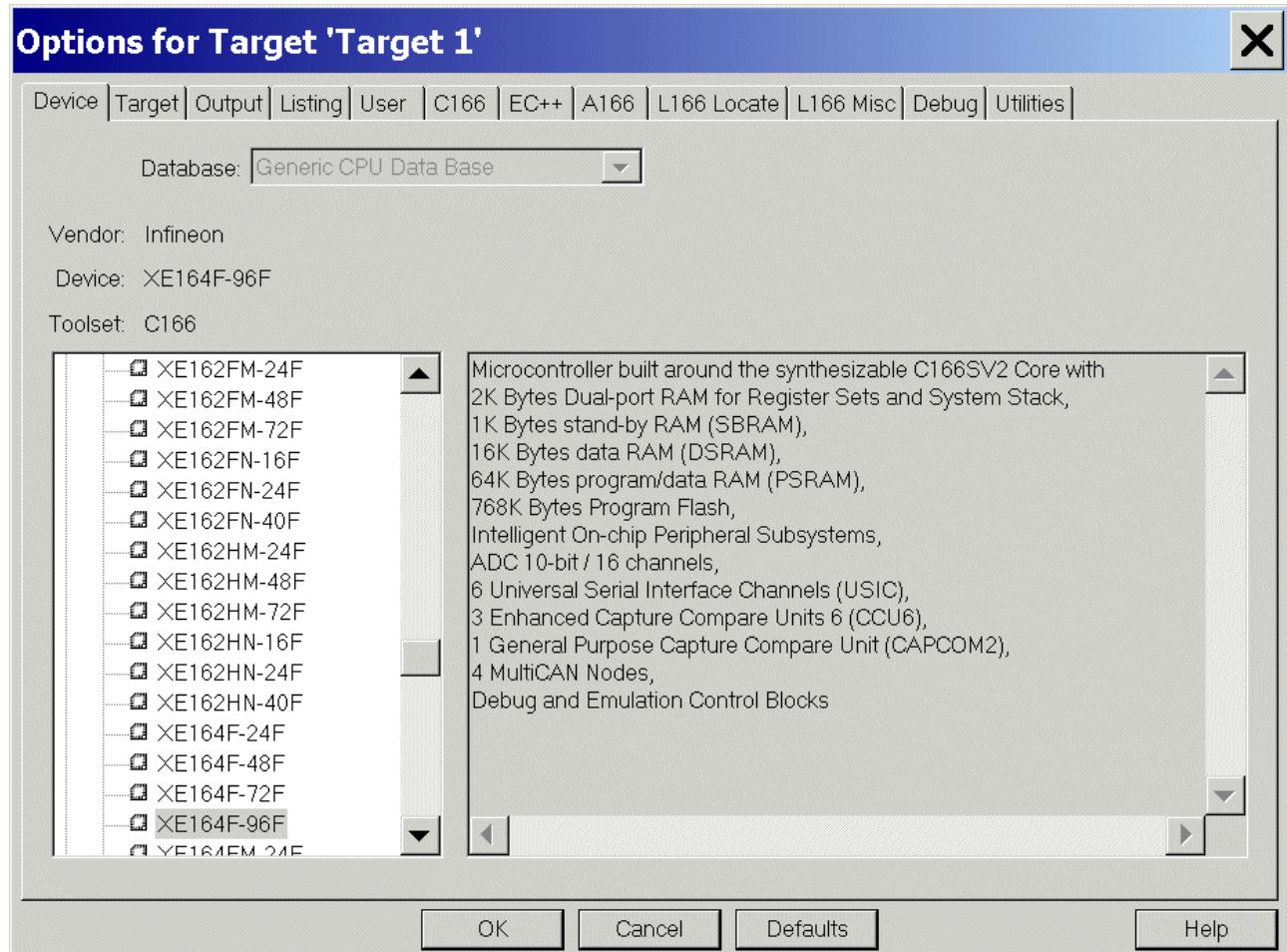


Configure:

Compiler, Assembler, Linker, Locater, Hex-Converter, Build – Control, (Simulator, Debugger,)  
Listings and Utilities (e.g. OnChip Flash Programming):



Device: check XE164F-96F



Target: Clock(MHz): **check** 8.0

Target: tick/check  Use On-chip ROM

Target: tick/check  Use On-chip ROM

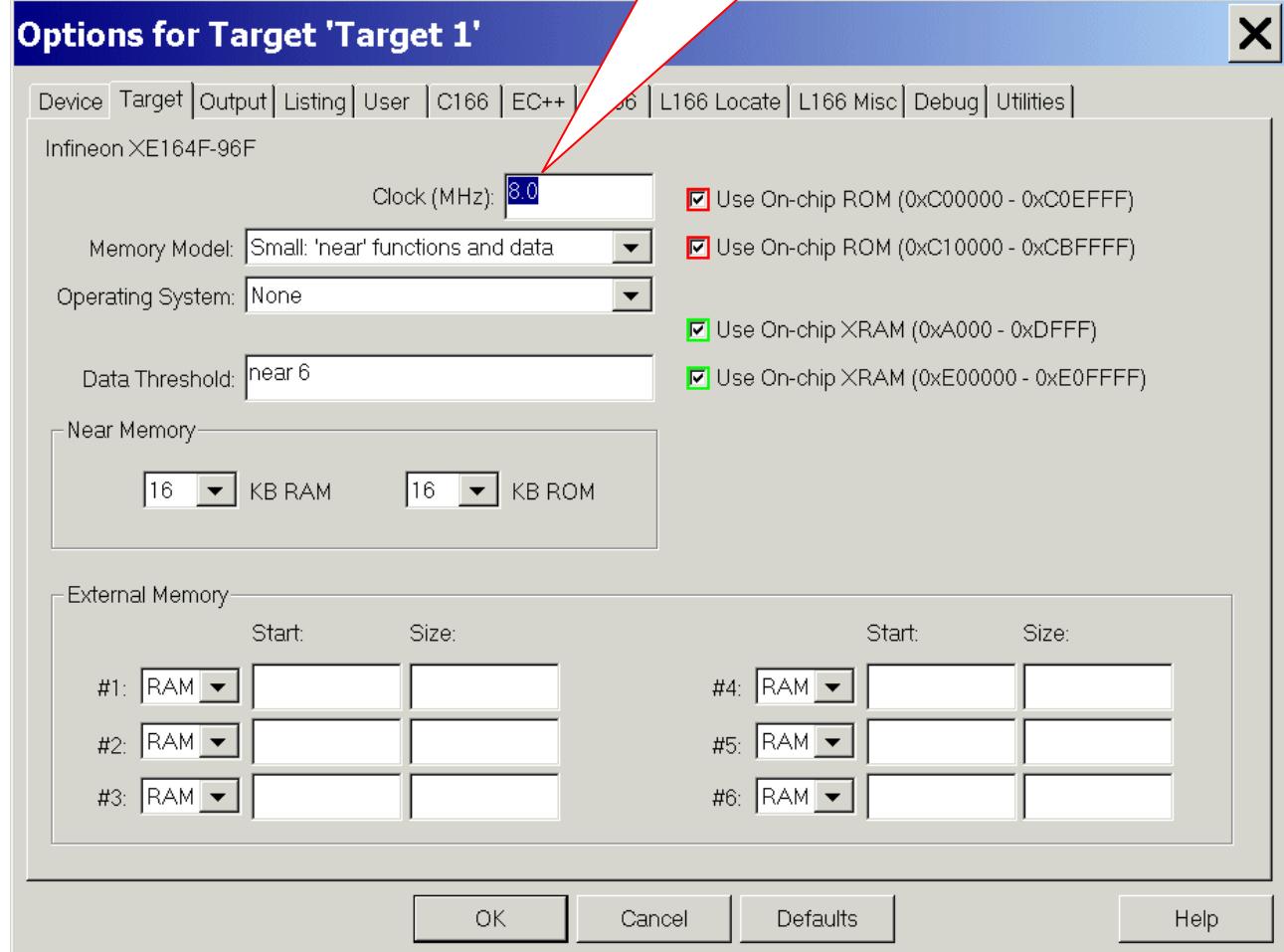
Target: tick/check  Use On-chip XRAM

Target: tick/check  Use On-chip XRAM

**Note (Source: DAvE):**

Configuration of the System Clock:

- VCO clock used, input clock is connected
- input frequency is 8,00 MHz
- configured system frequency is 66,00 MHz
- system clock is 66.00 MHz




Additional information: **Memory Map** (Source: User's Manual):

**Options for Target 'Target 1'**

Device	Target	Output	Listing	User	C166	EC++	A166	L166 Locate	L166 Misc	Debug	Utilities
Infineon XE164F-96F											
				Clock (MHz): <input type="text" value="8.0"/> <input checked="" type="checkbox"/> Use On-chip ROM (0xC00000 - 0xC0EFFF) Memory Model: Small: 'near' functions and data <input type="button" value="▼"/> <input checked="" type="checkbox"/> Use On-chip ROM (0xC10000 - 0xCBFFFF) Operating System: None <input type="button" value="▼"/> <input checked="" type="checkbox"/> Use On-chip XRAM (0xA000 - 0xDFFF) Data Threshold: near 6 <input checked="" type="checkbox"/> Use On-chip XRAM (0xE00000 - 0xE0FFFF)							

**Table 3-1 XE16x Memory Map<sup>1)</sup>**

Address Area	Start Loc.	End Loc.	Area Size <sup>2)</sup>	Notes
IMB register space	FF'FF00 <sub>H</sub>	FF'FFFF <sub>H</sub>	256 Bytes	
Reserved (access trap)	F0'0000 <sub>H</sub>	FF'FEFF <sub>H</sub>	< 1 MByte	Minus IMB registers.
Reserved for EPSRAM	E9'0000 <sub>H</sub>	EF'FFFF <sub>H</sub>	448 KBytes	
EPSRAM	E8'0000 <sub>H</sub>	E8'FFFF <sub>H</sub>	64 KBytes	PSRAM with Flash timing.
Reserved for PSRAM	E1'0000 <sub>H</sub>	E7'FFFF <sub>H</sub>	448 KBytes	
PSRAM	E0'0000 <sub>H</sub>	E0'FFFF <sub>H</sub>	64 KBytes	Program SRAM.
Reserved for Flash	CC'0000 <sub>H</sub>	DF'FFFF <sub>H</sub>	<1.25 MBytes	
Flash 2	C8'0000 <sub>H</sub>	CB'FFFF <sub>H</sub>	256 KBytes	
Flash 1	C4'0000 <sub>H</sub>	C7'FFFF <sub>H</sub>	256 KBytes	
Flash 0	C0'0000 <sub>H</sub>	C3'FFFF <sub>H</sub>	252 KBytes <sup>3)</sup>	Minus res. seg.
External memory area	40'0000 <sub>H</sub>	BF'FFFF <sub>H</sub>	8 MBytes	
External IO area <sup>4)</sup>	20'5800 <sub>H</sub>	3F'FFFF <sub>H</sub>	< 2 MBytes	Minus CAN/USIC
USIC registers	20'4000 <sub>H</sub>	20'57FF <sub>H</sub>	6 KBytes	Accessed via EBC
MultiCAN registers	20'0000 <sub>H</sub>	20'3FFF <sub>H</sub>	16 KBytes	Accessed via EBC
External memory area	01'0000 <sub>H</sub>	1F'FFFF <sub>H</sub>	< 2 MBytes	Minus segment 0
SFR area	00'FE00 <sub>H</sub>	00'FFFF <sub>H</sub>	0.5 KBytes	
Dual-port RAM (DPRAM)	00'F600 <sub>H</sub>	00'FDFF <sub>H</sub>	2 KBytes	
Reserved for DPRAM	00'F200 <sub>H</sub>	00'F5FF <sub>H</sub>	1 KBytes	
ESFR area	00'F000 <sub>H</sub>	00'F1FF <sub>H</sub>	0.5 KBytes	
XSFR area	00'E000 <sub>H</sub>	00'EFFF <sub>H</sub>	4 KBytes	
Data SRAM (DSRAM)	00'A000 <sub>H</sub>	00'DFFF <sub>H</sub>	16 KBytes	
Reserved for DSRAM	00'8000 <sub>H</sub>	00'9FFF <sub>H</sub>	8 KBytes	
External memory area	00'0000 <sub>H</sub>	00'7FFF <sub>H</sub>	32 KBytes	

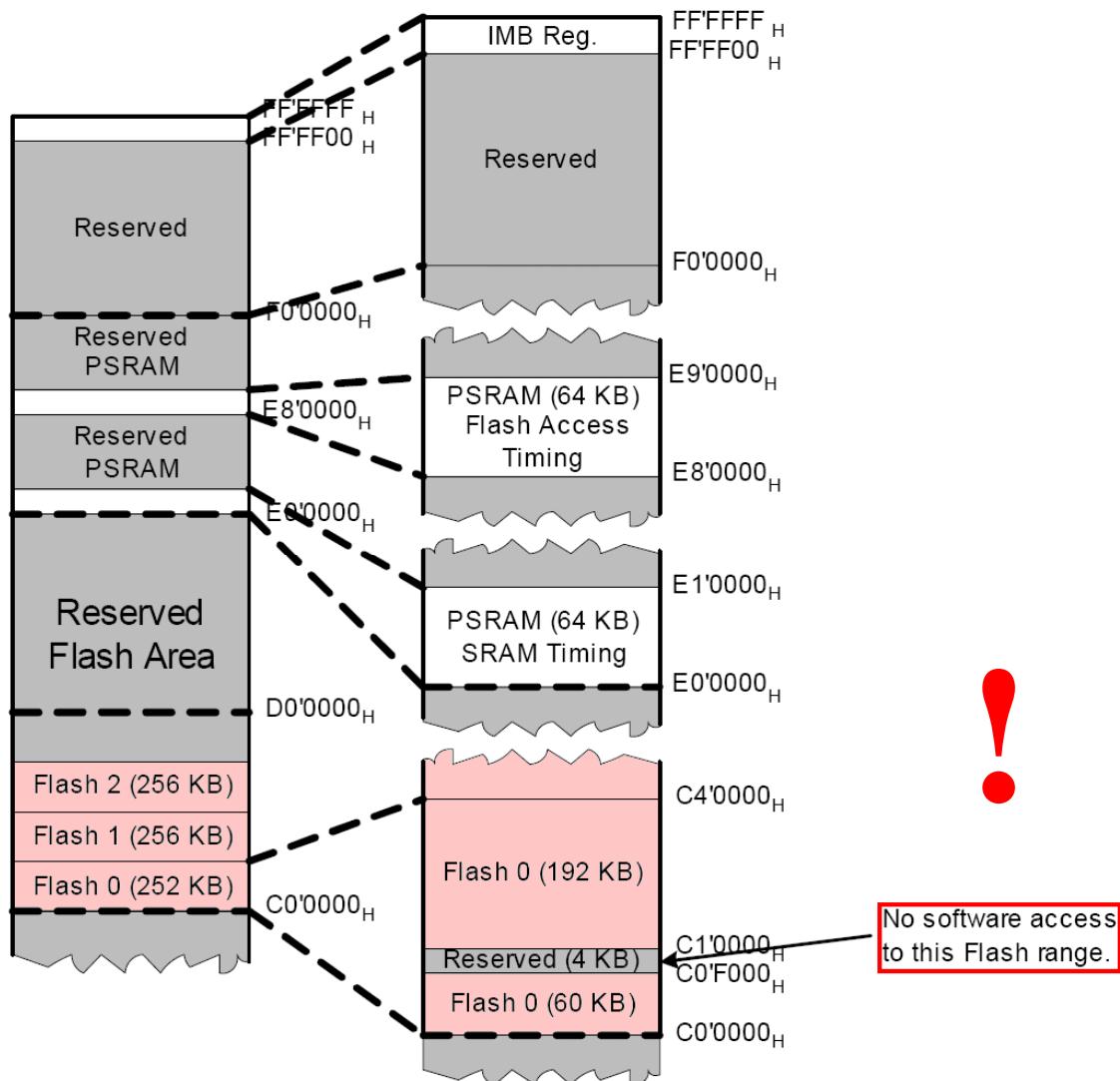

Additional information: **Memory Map** (Source: User's Manual):

**Options for Target 'Target 1'**

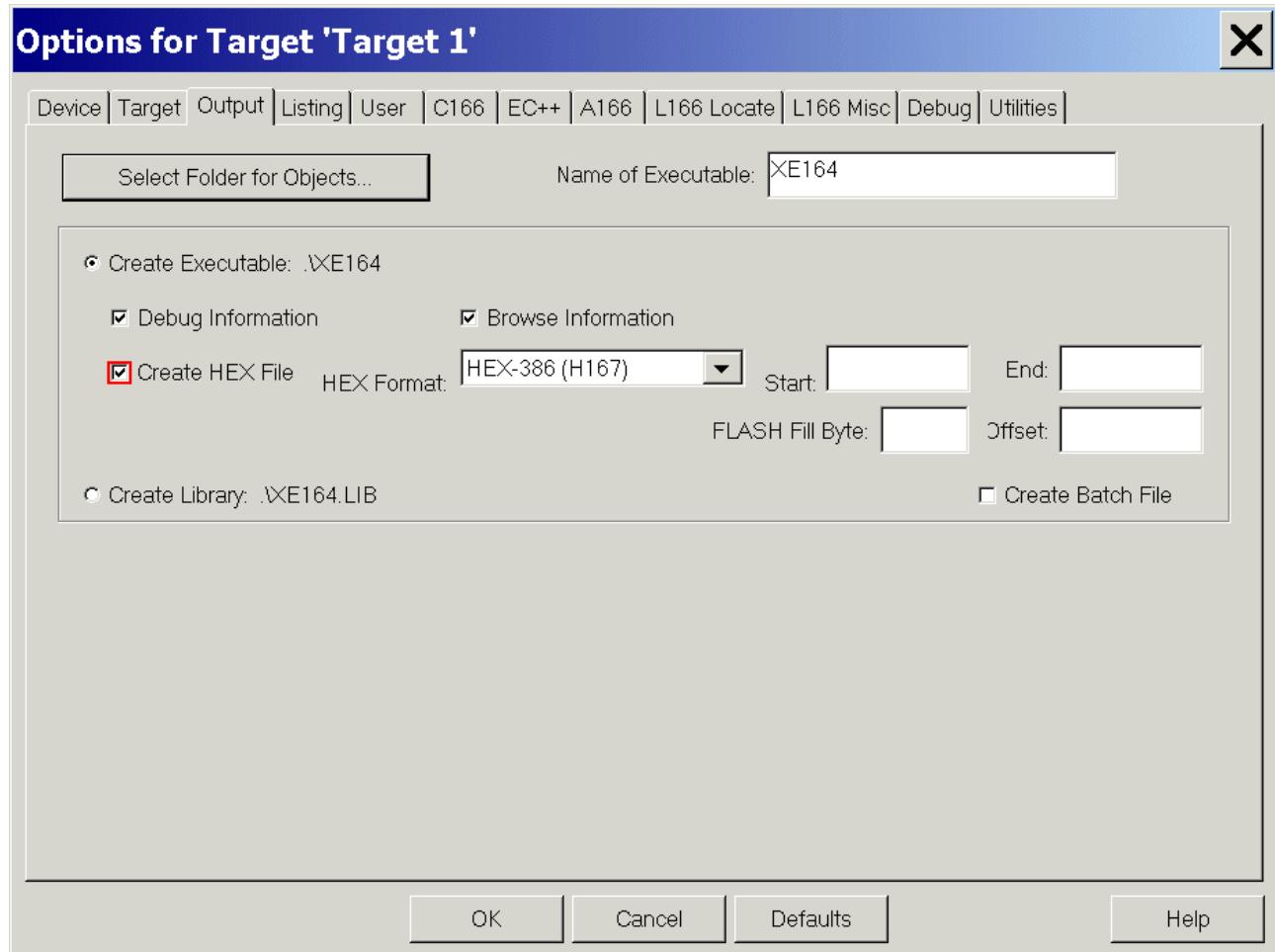
Device | Target | Output | Listing | User | C166 | EC++ | A166 | L166 Locate | L166 Misc | Debug | Utilities |

Infineon XE164F-96F

Clock (MHz):	8.0	<input checked="" type="checkbox"/> Use On-chip ROM (0xC00000 - 0xC0EFFF)
Memory Model:	Small: 'near' functions and data	<input checked="" type="checkbox"/> Use On-chip ROM (0xC10000 - 0xCBFFFF)
Operating System:	None	<input checked="" type="checkbox"/> Use On-chip XRAM (0xA000 - 0xDFFF)
Data Threshold:	near 6	<input checked="" type="checkbox"/> Use On-chip XRAM (0xE00000 - 0xE0FFFF)



Output: tick  Create HEX File

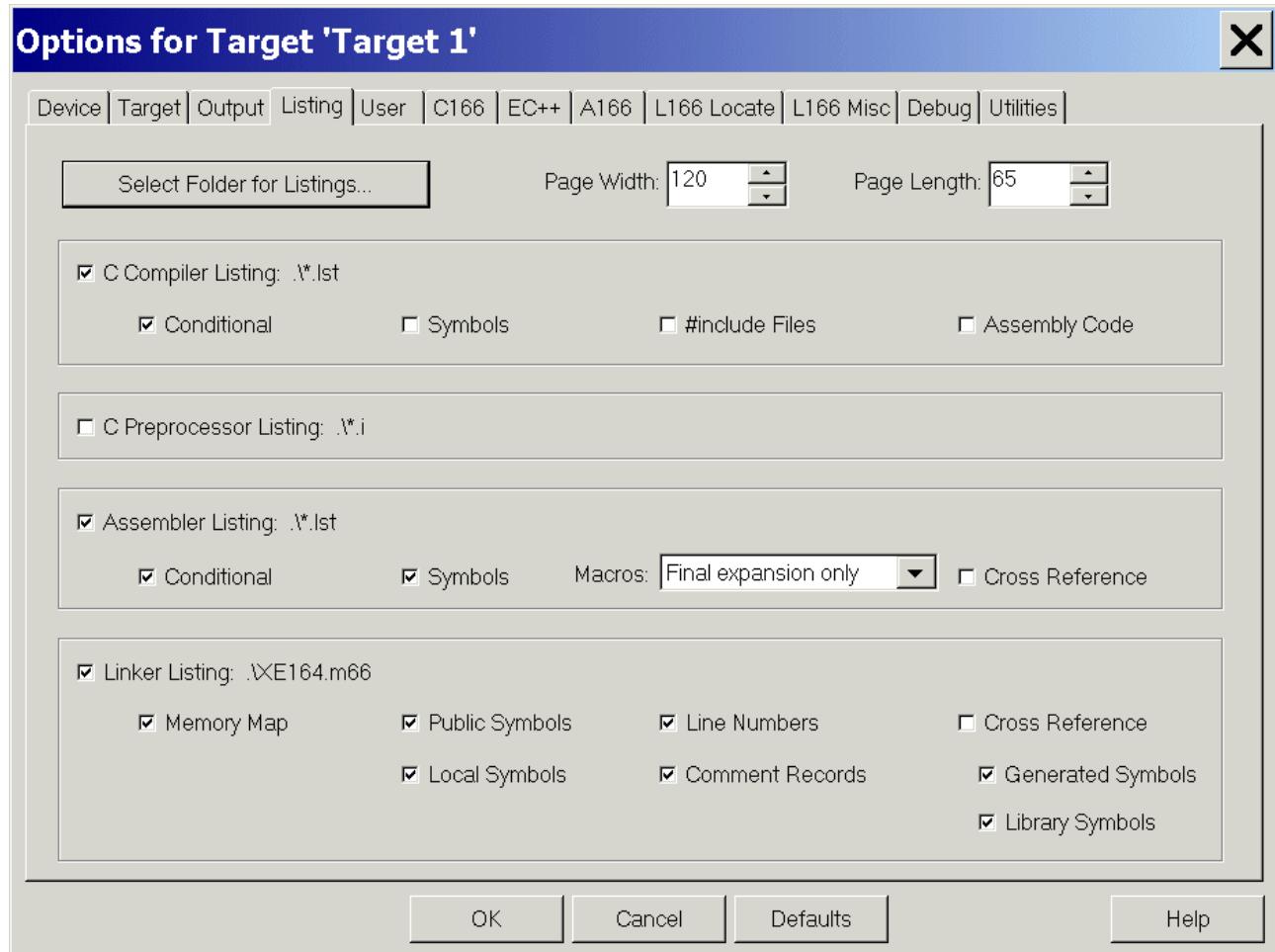


**Note:**

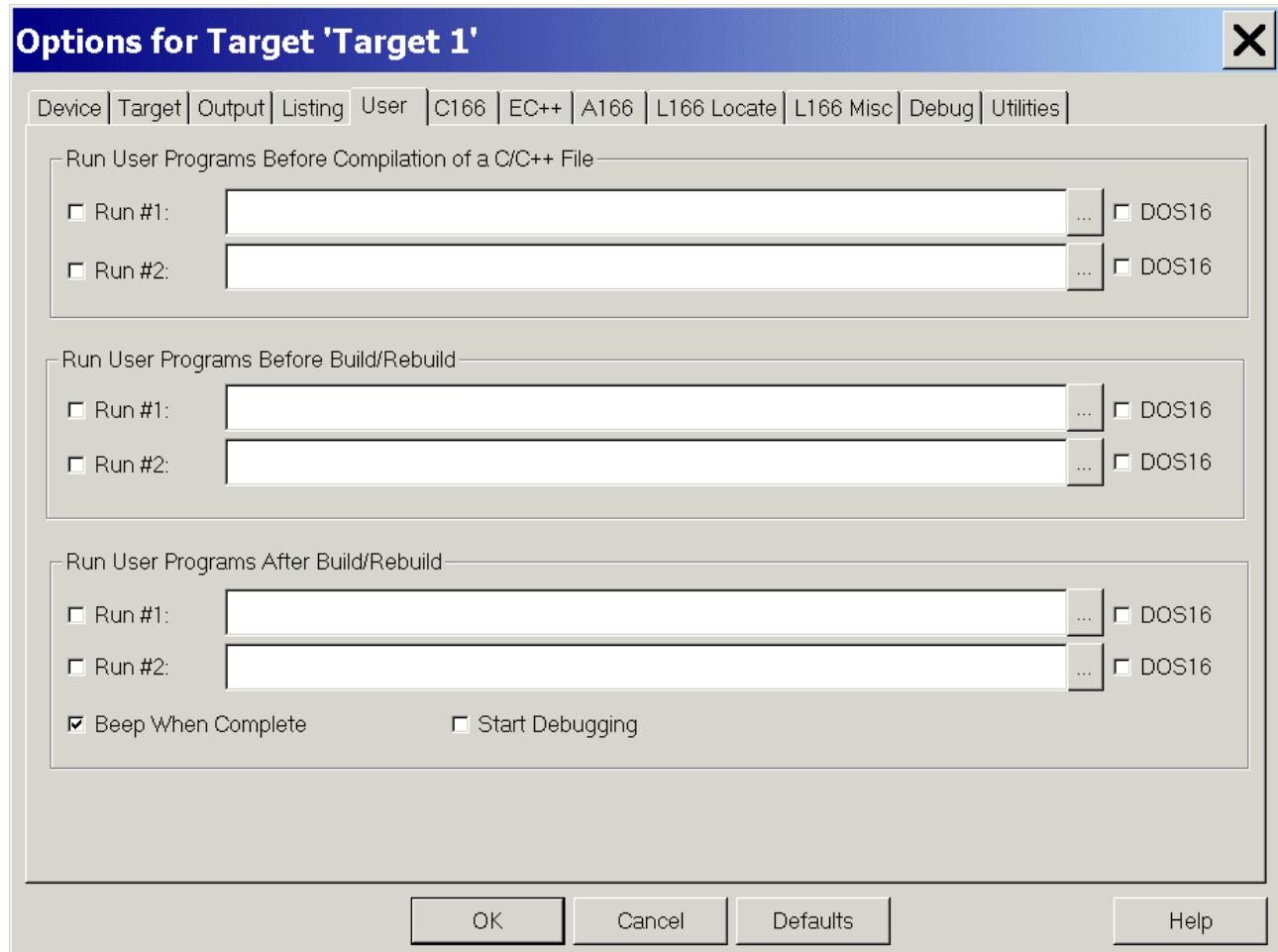
The HEX File could be used while working with the program  
MEMTOOL for OnChip-Flash-Programming  
via RS232-interface [Bootstrap Loader (BSL) Mode via UART/USIC0\_CH0].



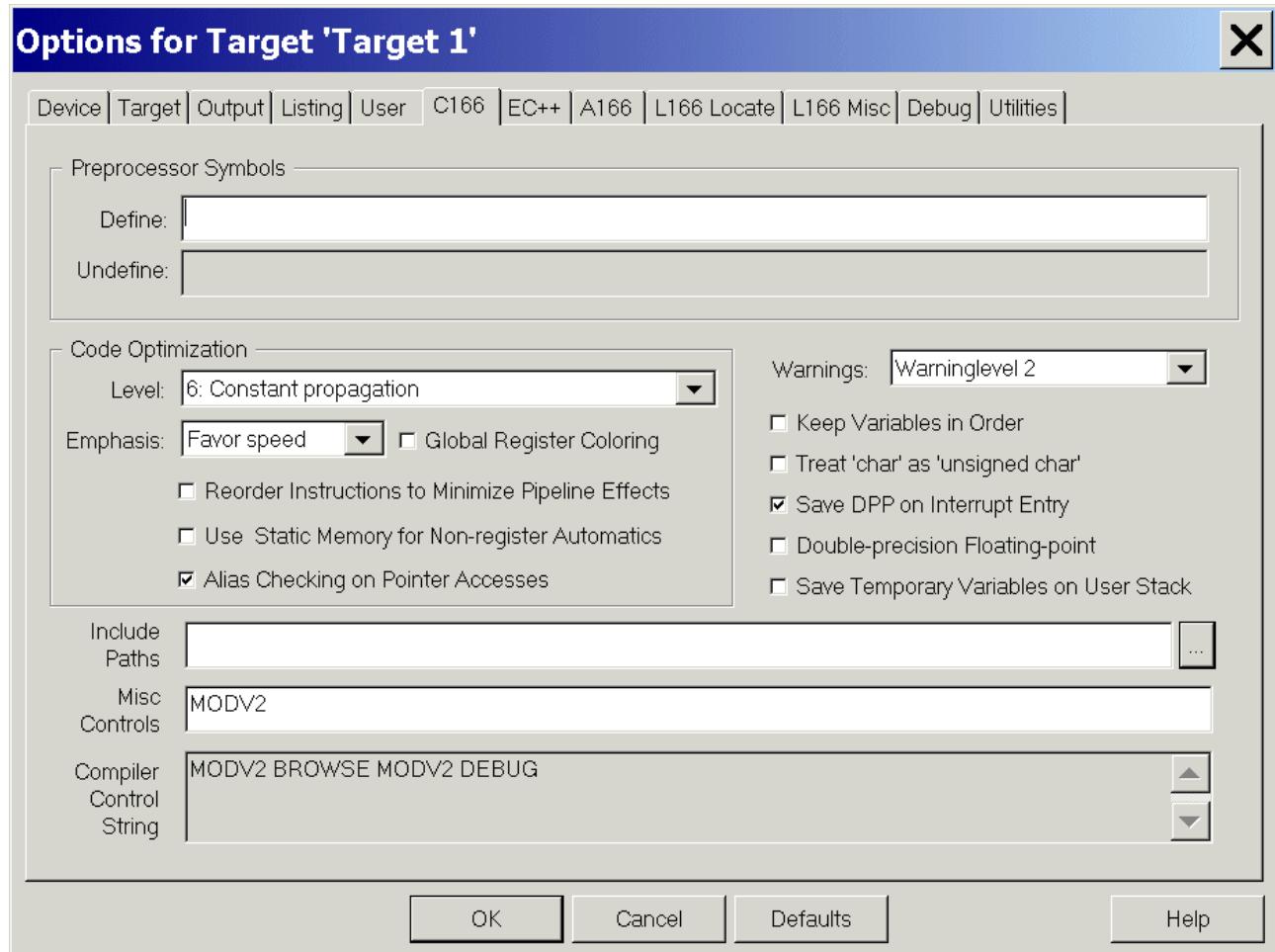
**Listing:** (do nothing)



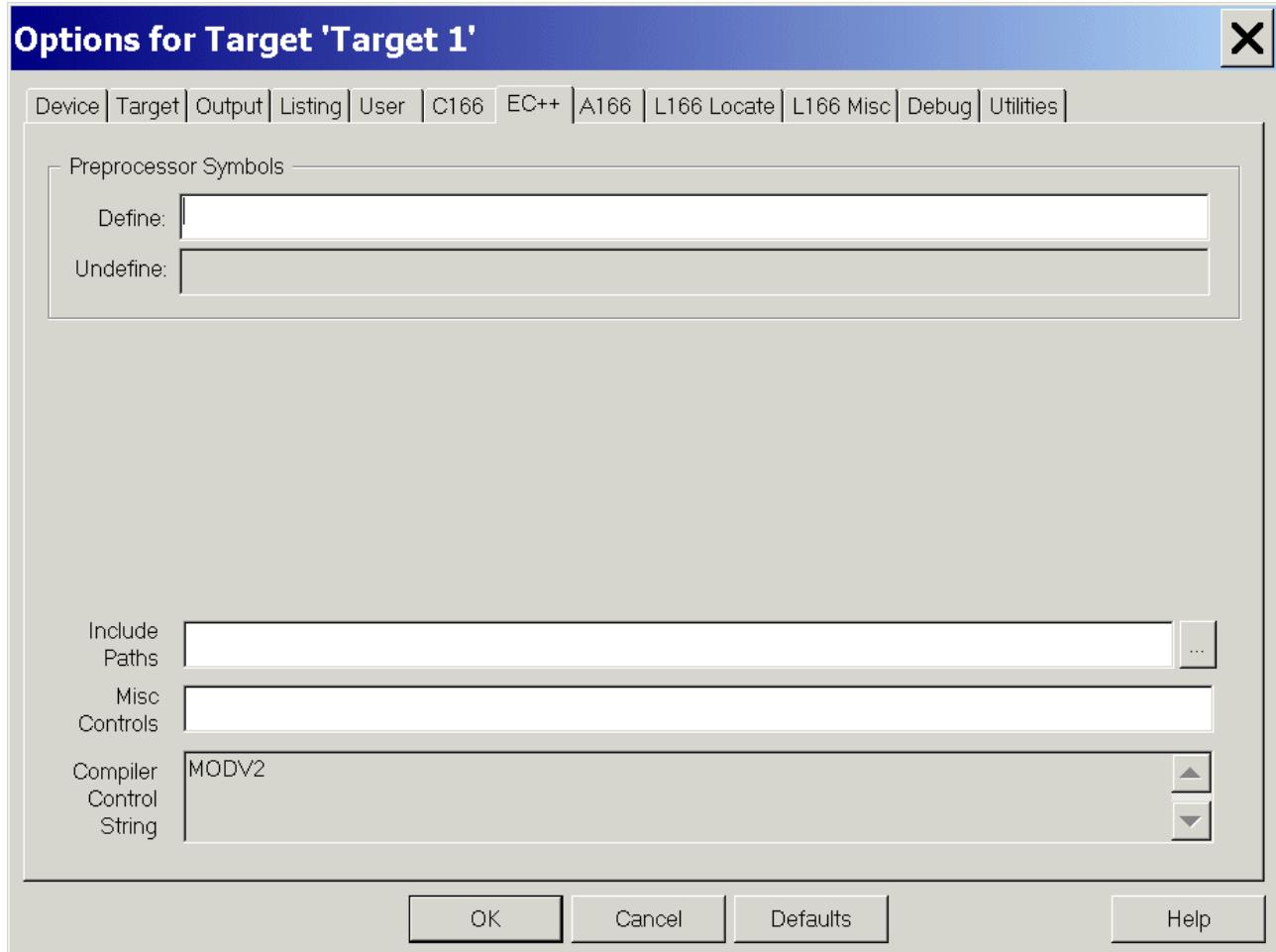
User: (do nothing)



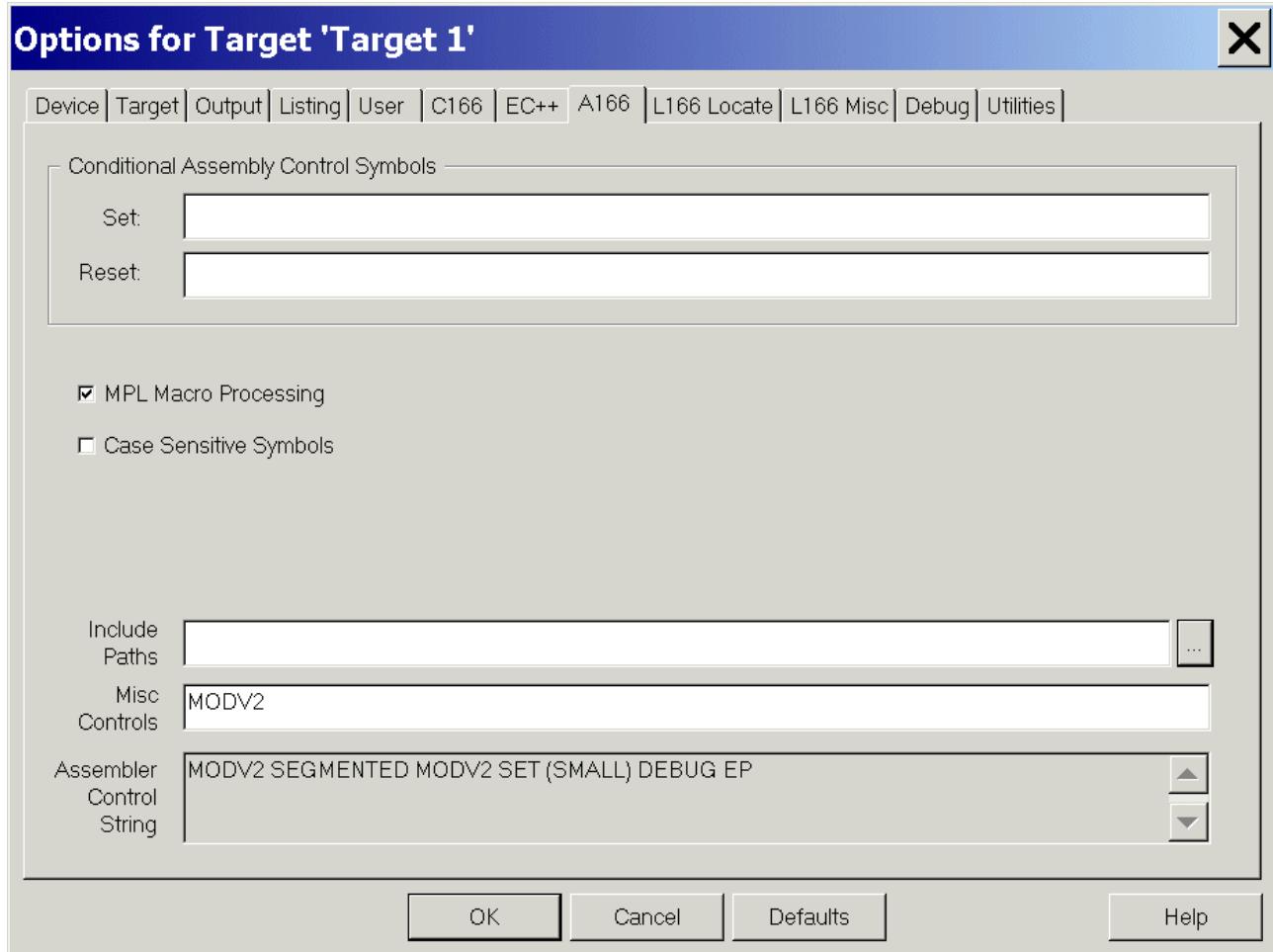
C166: (do nothing)



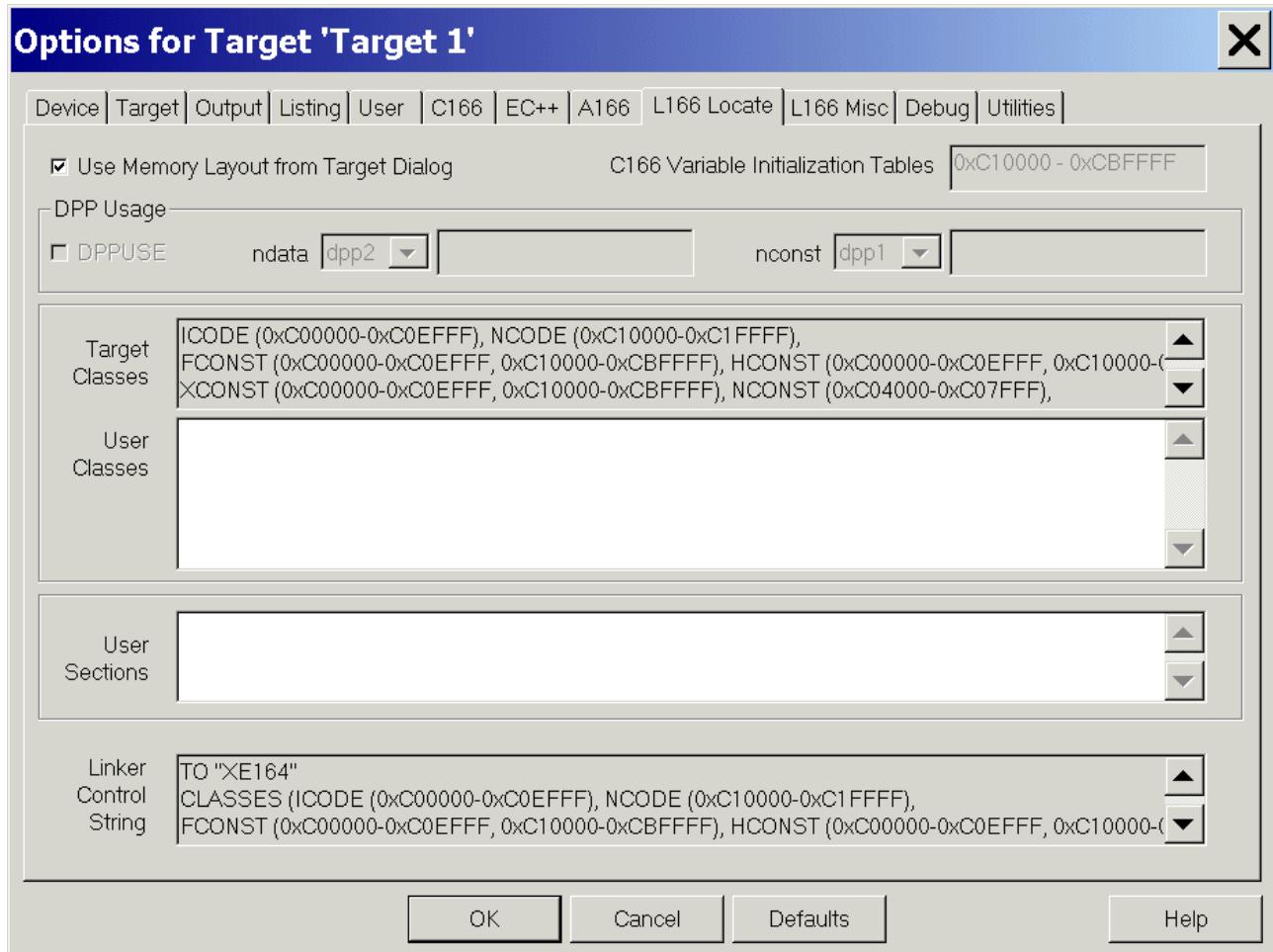
EC++: (do nothing)



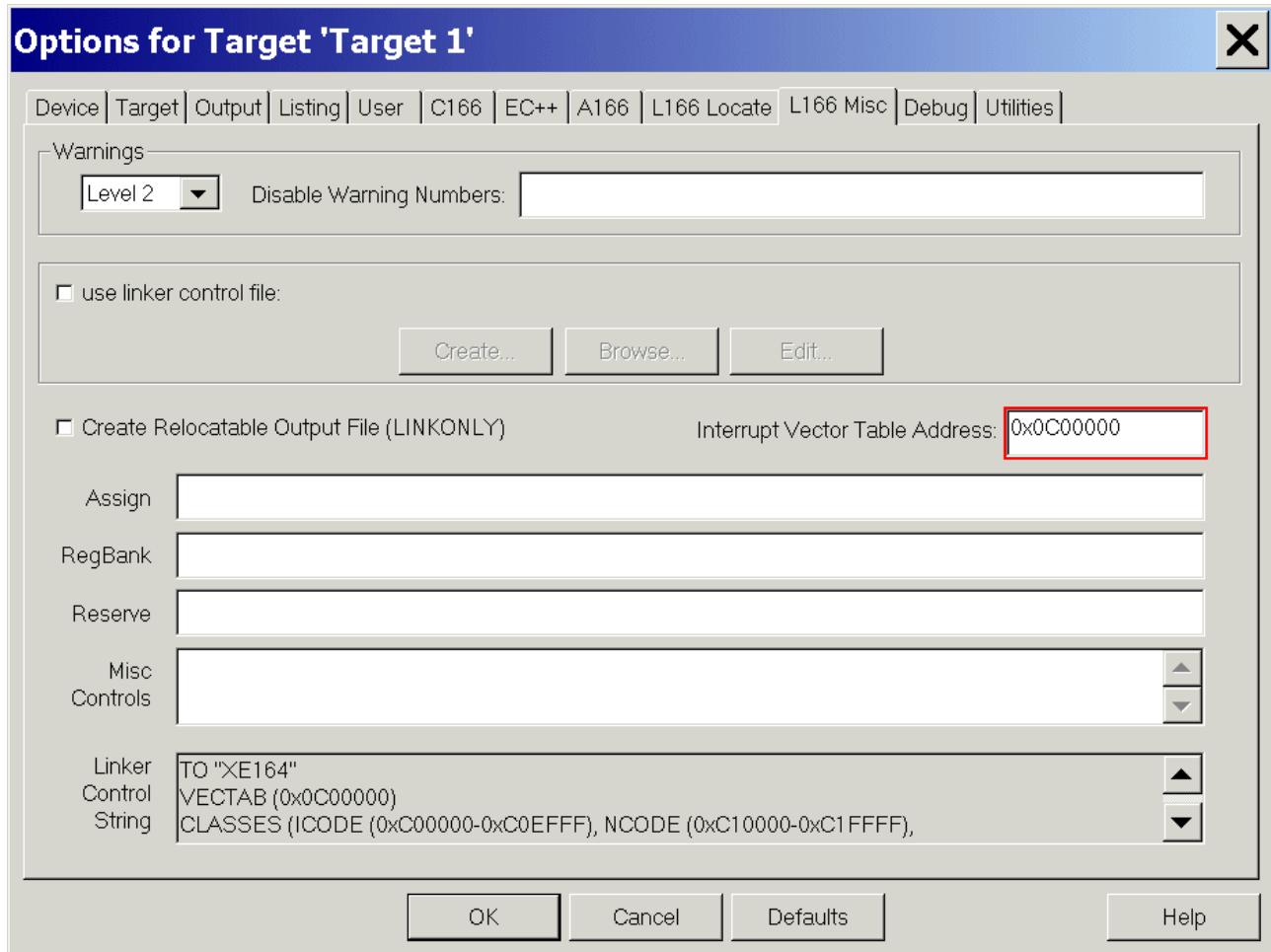
A166: (do nothing)



L166 Locate: (do nothing)

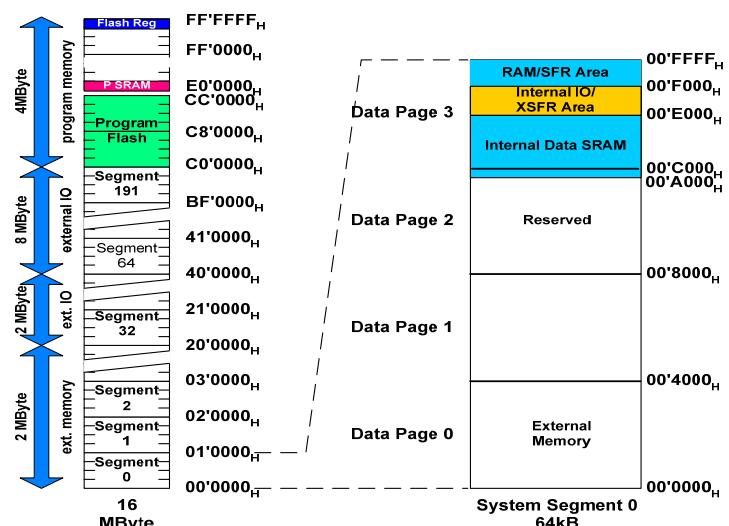


L166 Misc: Interrupt Vector Table Address: insert 0x0C00000

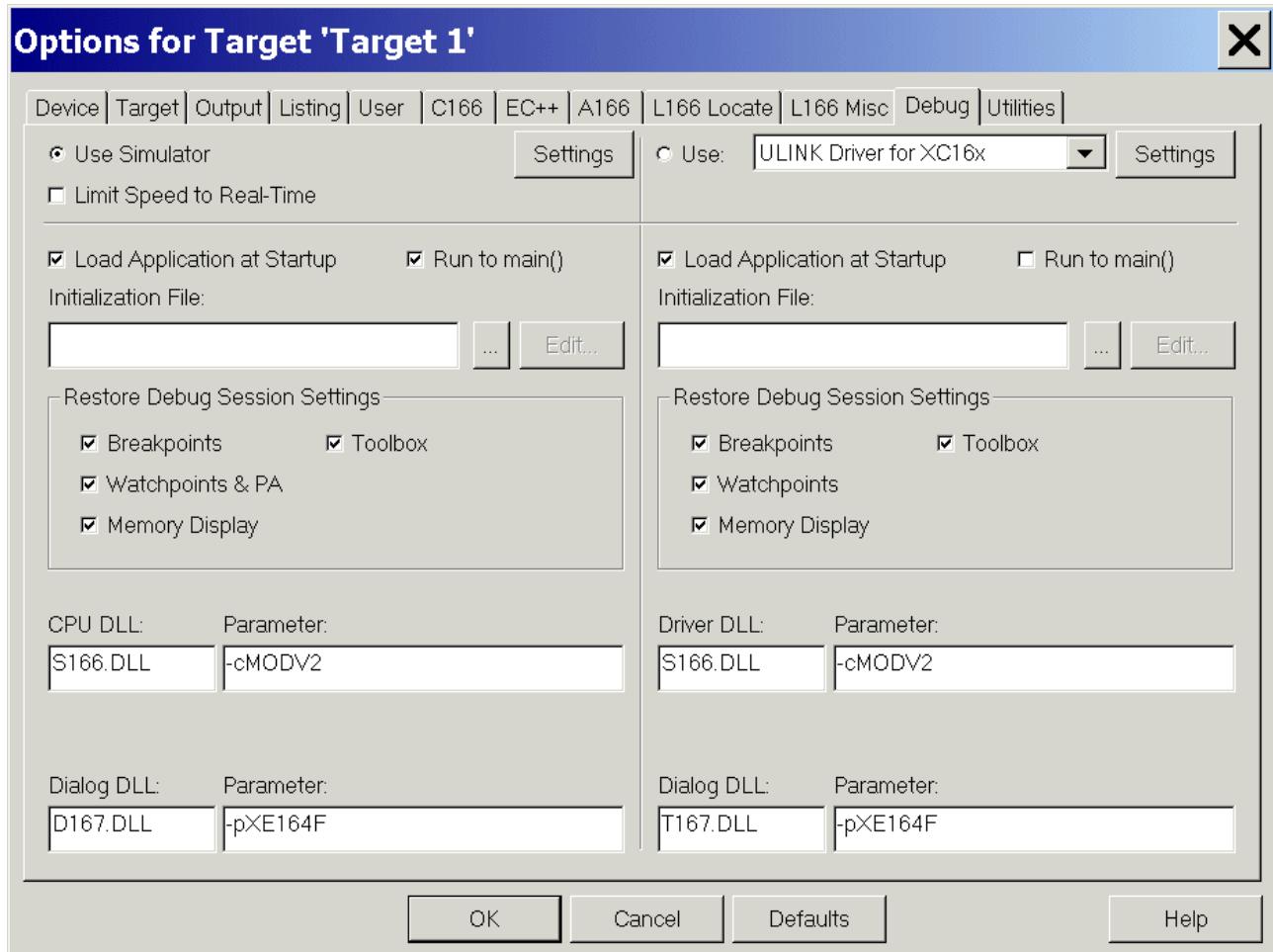


Note:

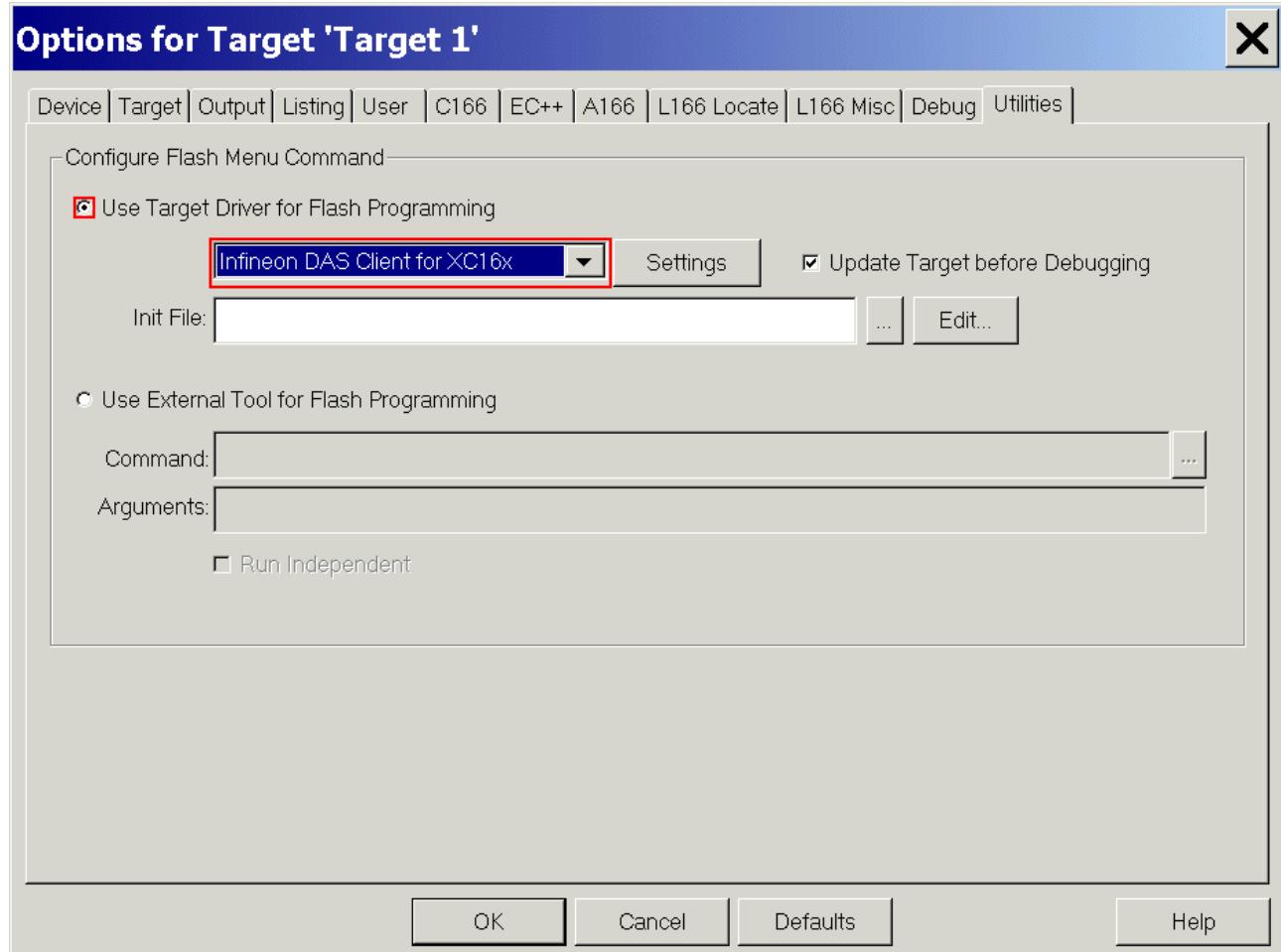
The On Chip Flash starts here.



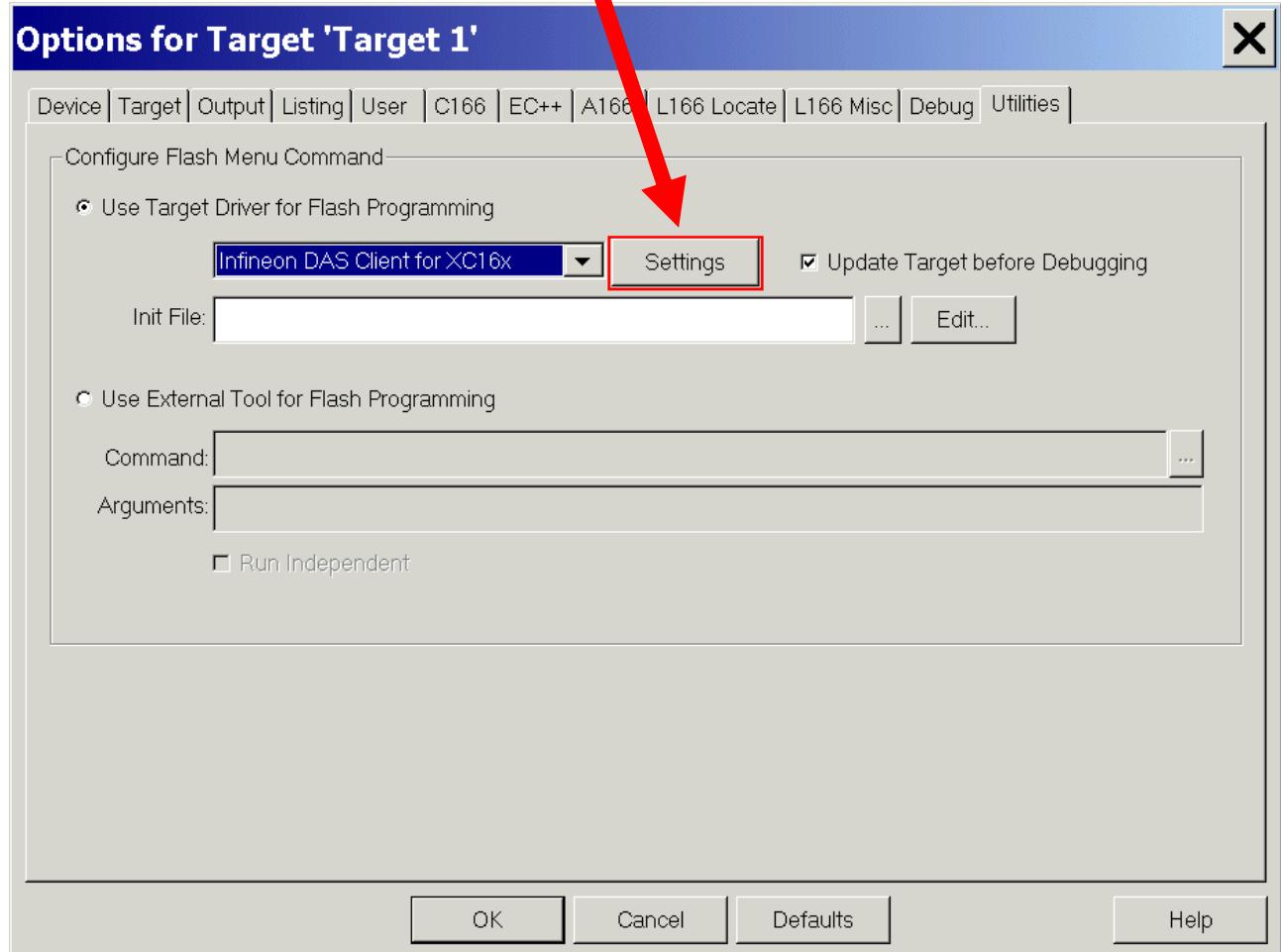
**Debug:** (do nothing)



Utilities: Configure Flash Menu Command: **check**  Use Target Driver for Flash Programming  
Utilities: Configure Flash Menu Command: **select** Infineon DAS Client for XC16x



Utilities: Configure Flash Menu Command: click Settings



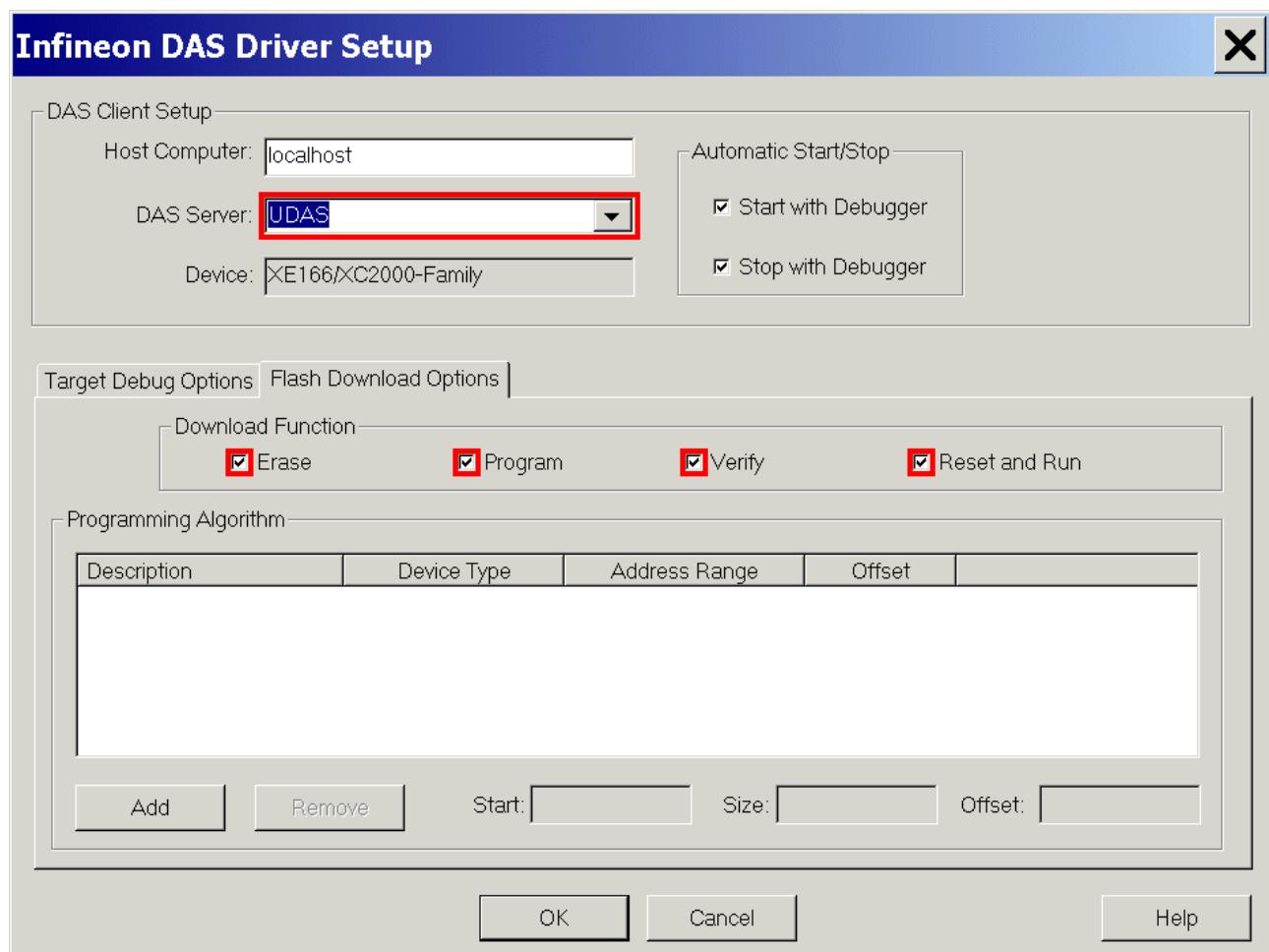
DAS Client Setup: DAS Server: select: UDAS (Universal Device Access Server)

Flash Download Options: Download Function: tick/check:  Erase

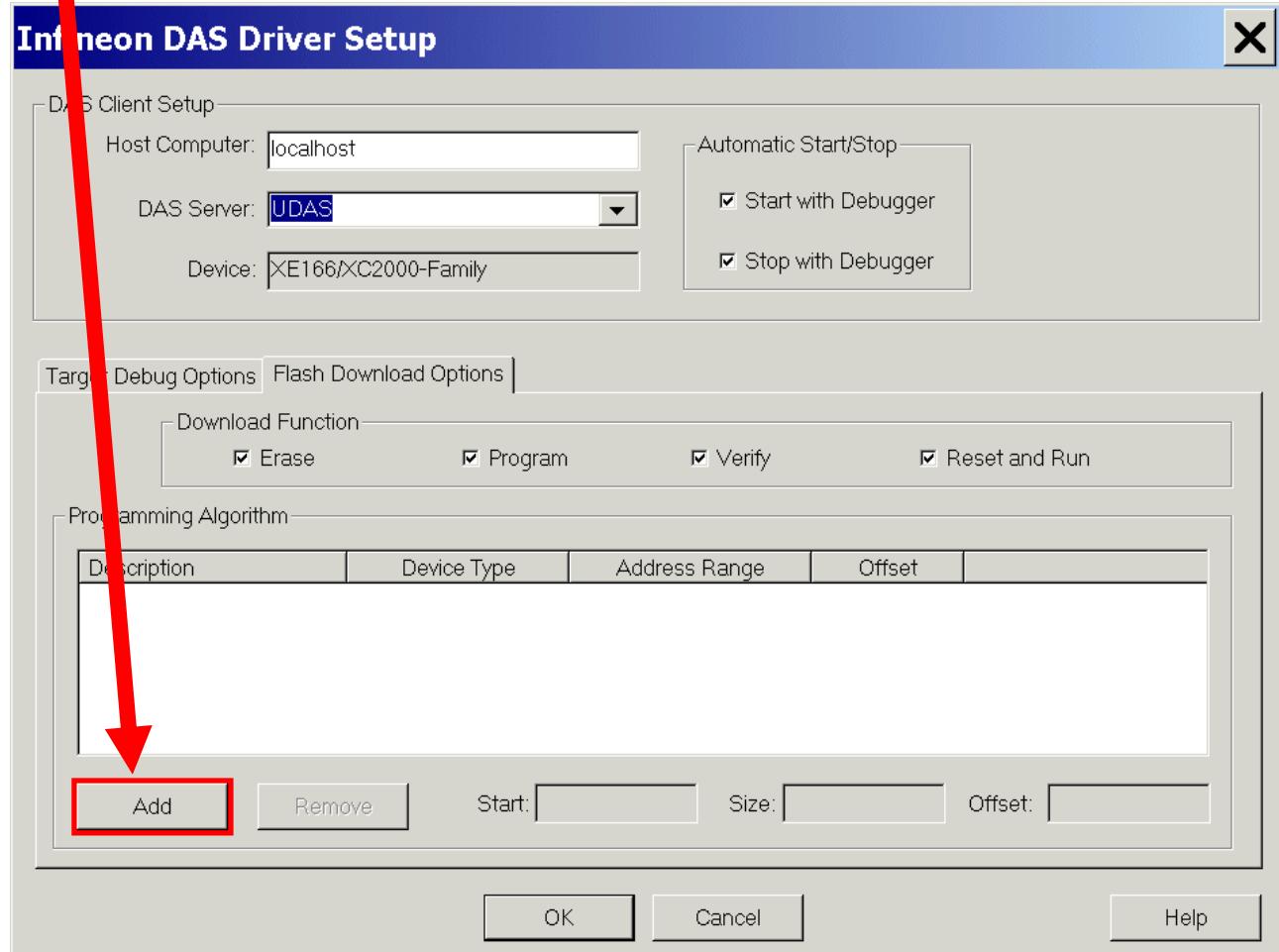
Flash Download Options: Download Function: tick/check:  Program

Flash Download Options: Download Function: tick/check:  Verify

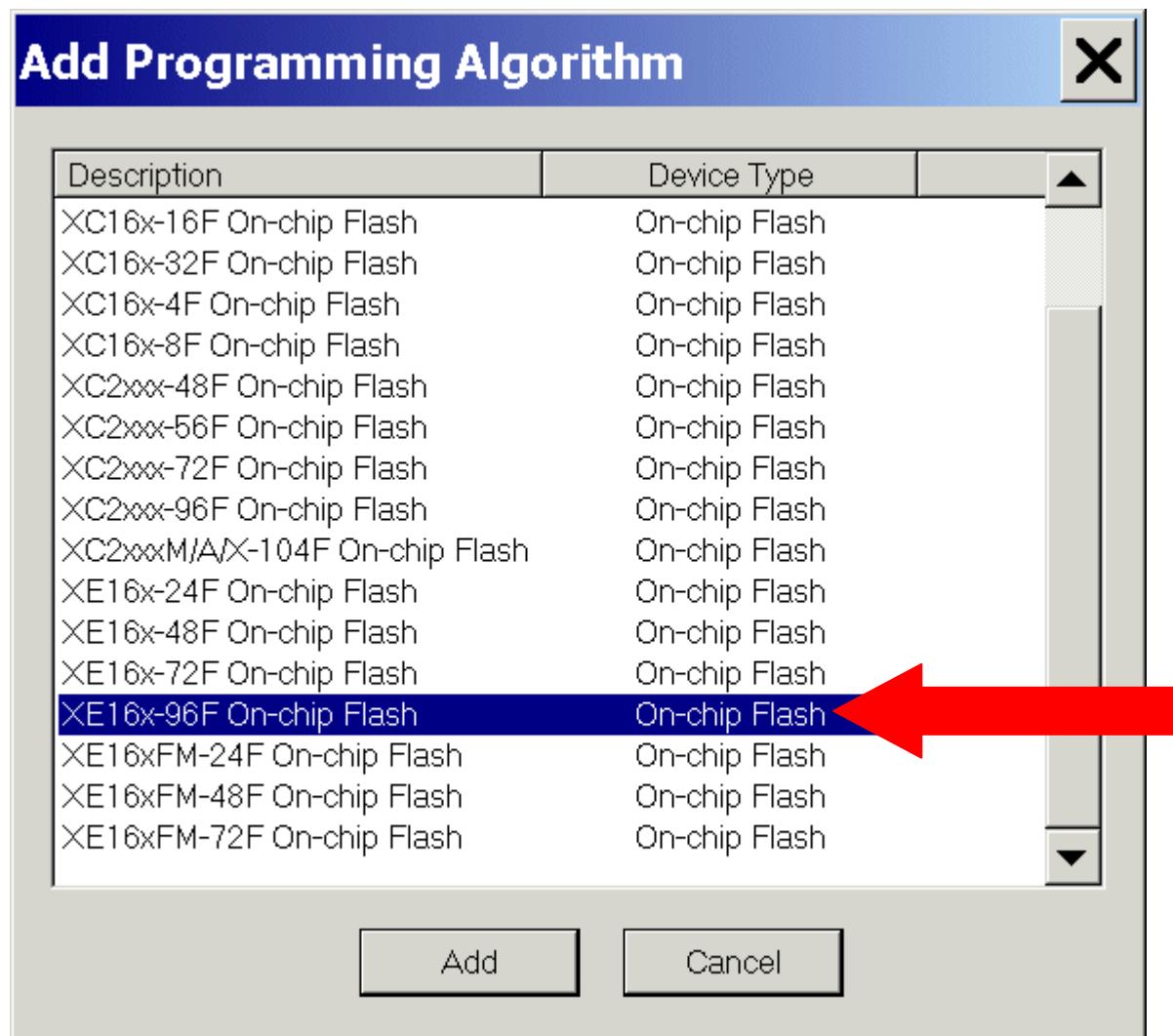
Flash Download Options: Download Function: tick/check:  Reset and Run



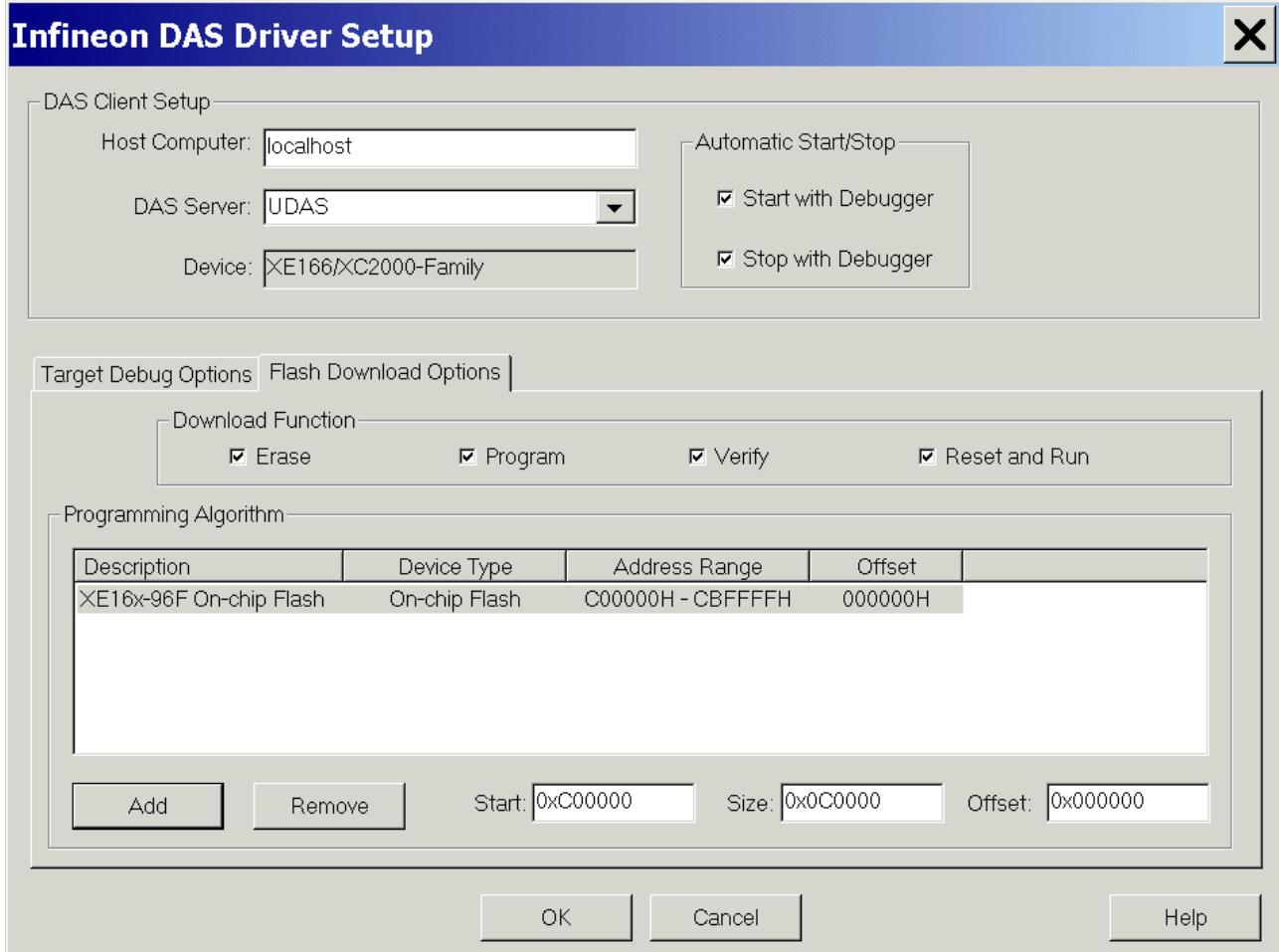
Click Add



Select: XE16x-96F On-chip Flash

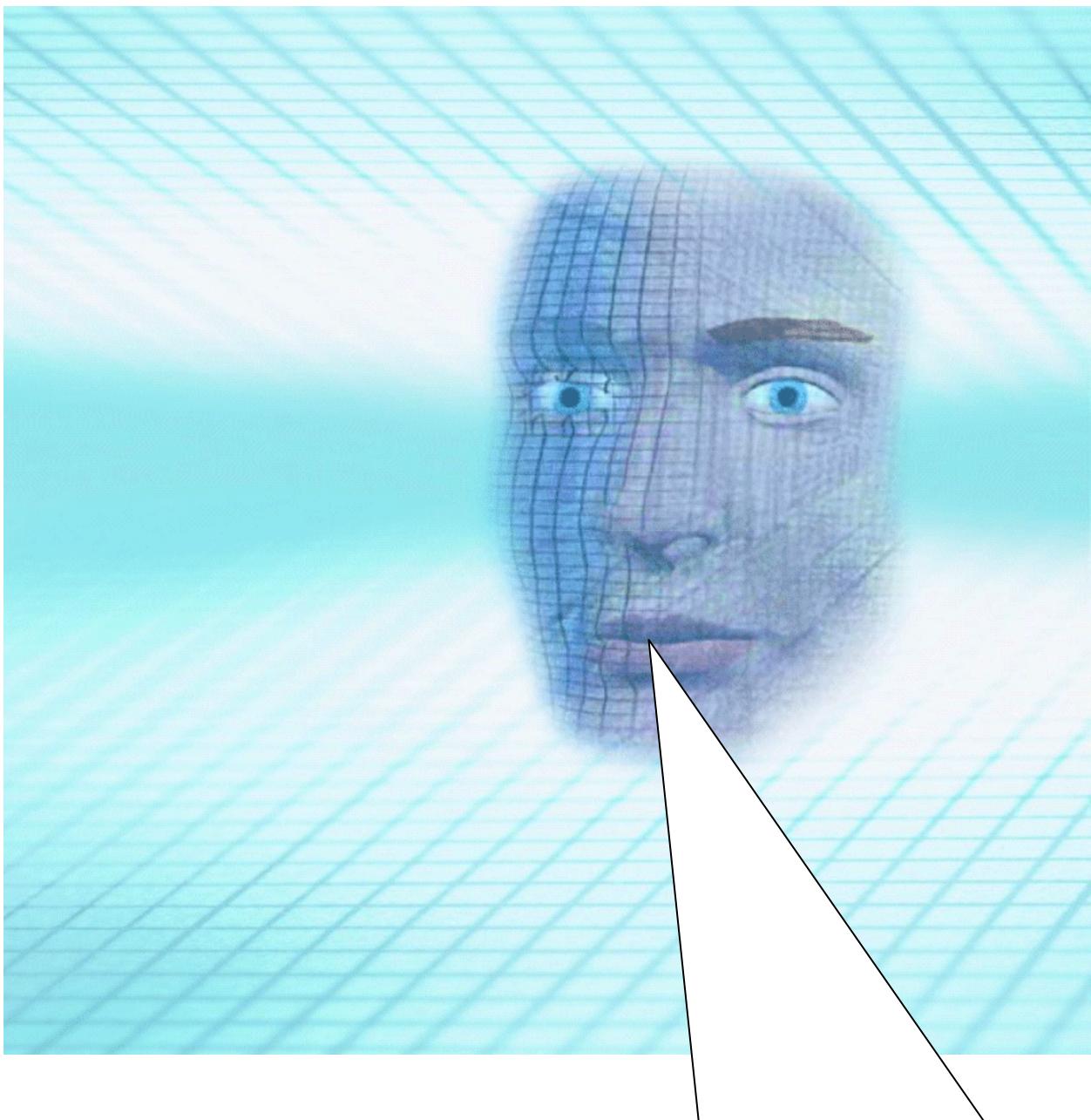


Click 



**OK**  
**OK**

Insert your application specific program:



Note:

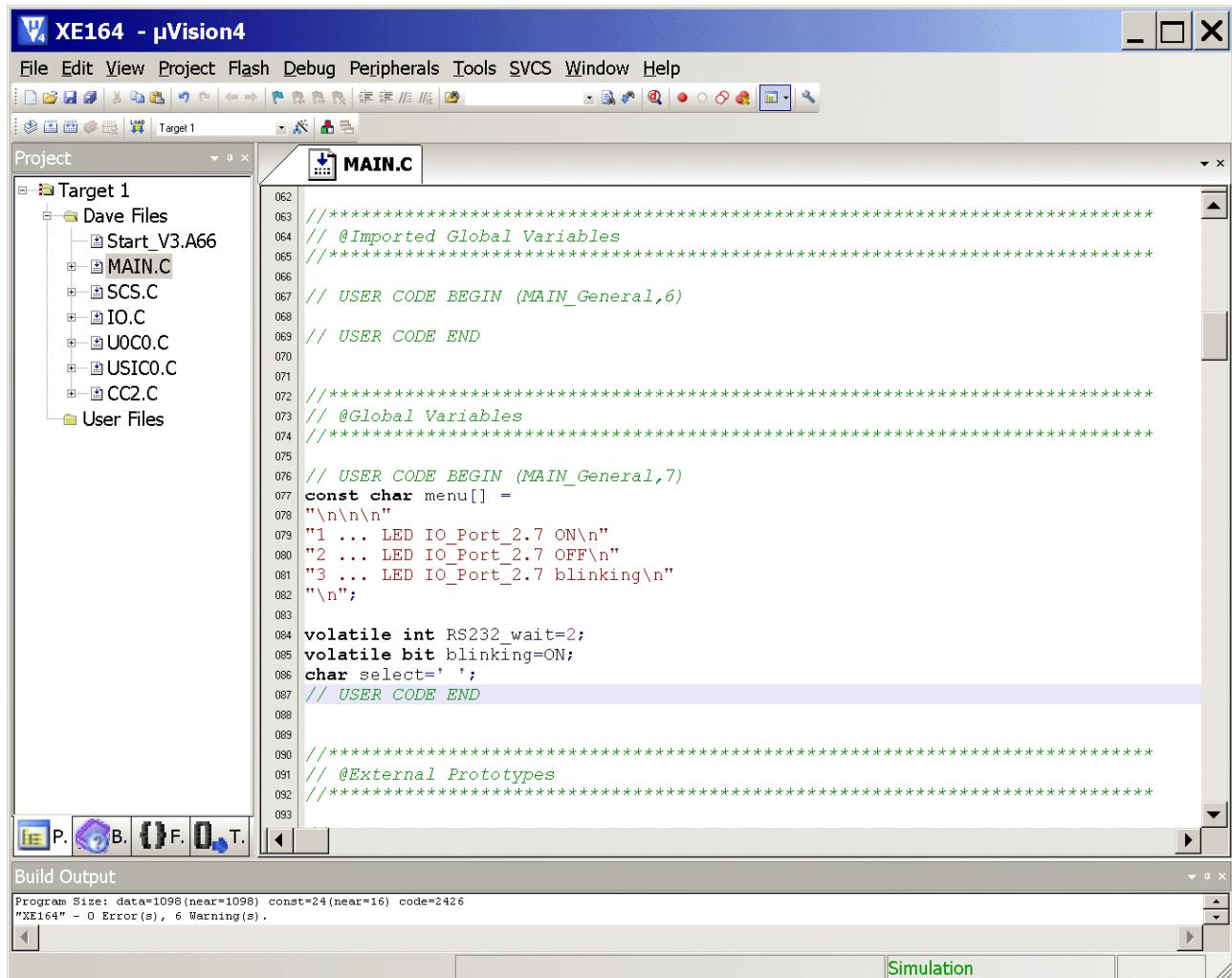
DAvE doesn't change code which is inserted between '`// USER CODE BEGIN`' and '`// USER CODE END`'. Therefore, whenever adding code to DAvE's generated code, write it between '`// USER CODE BEGIN`' and '`// USER CODE END`'.

If you wish to change DAvE's generated code or add code outside these 'USER CODE' sections you will have to insert/modify your changes each time after letting DAvE regenerate code!

Double click MAIN.C and insert Global Variables:

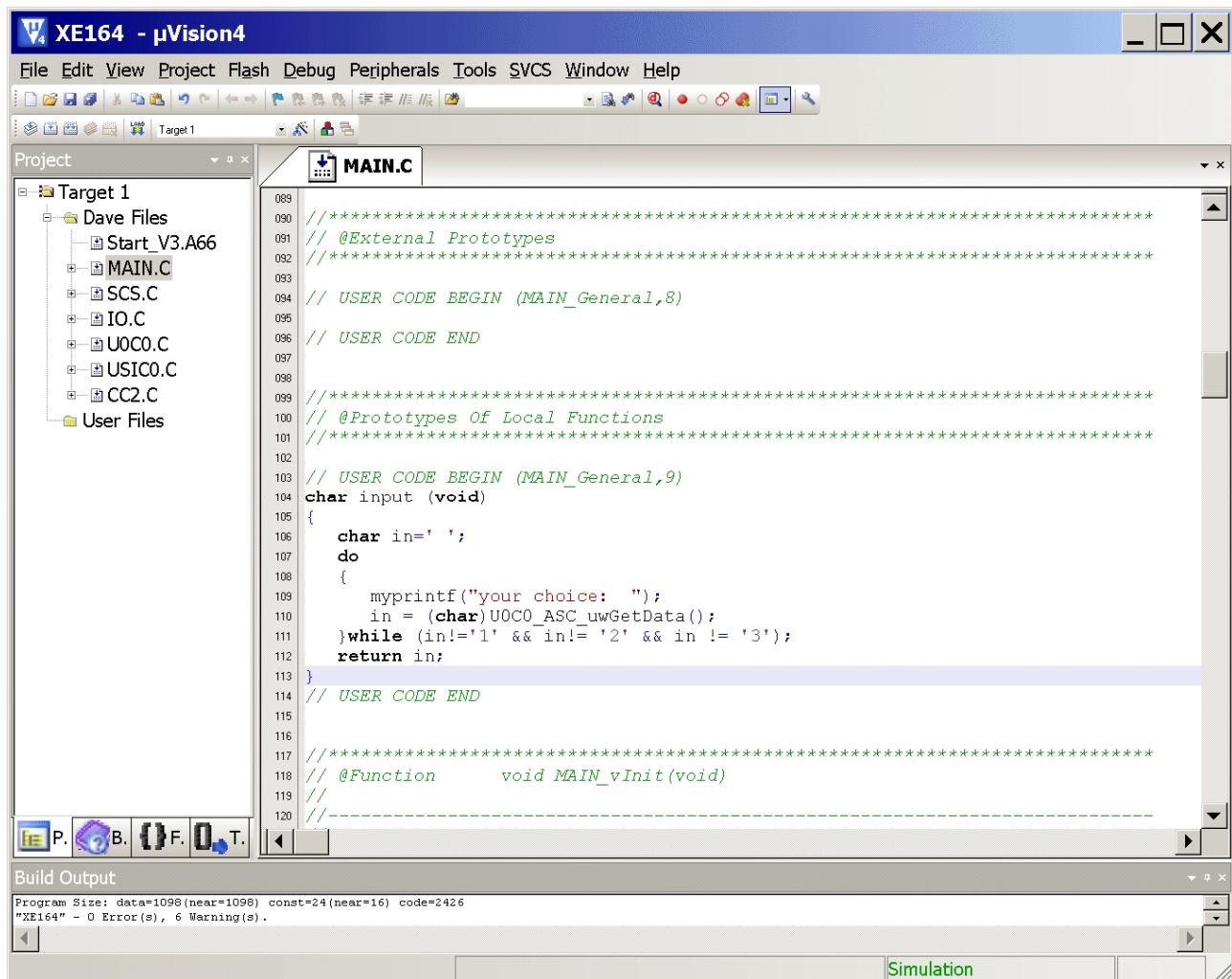
```
const char menu[] =
"\n\n\n"
"1 ... LED IO_Port_2.7 ON\n"
"2 ... LED IO_Port_2.7 OFF\n"
"3 ... LED IO_Port_2.7 blinking\n"
"\n";

volatile int RS232_wait=2;
volatile bit blinking=ON;
char select=' ';
```



Double click MAIN.C and insert the function input():

```
char input (void)
{
    char in=' ';
    do
    {
        myprintf("your choice: ");
        in = (char)U0C0_ASC_uwGetData();
    }while (in!='1' && in!= '2' && in != '3');
    return in;
}
```



Double click MAIN.C and insert the following code in the main function:

```
while(RS232_wait);
```

```

353 //-
354 //-----@Date 13.04.2010
355 //-----
356 //*****USER CODE BEGIN (Main,1)
357 //*****USER CODE END
358
359 void main(void)
360 {
361     // USER CODE BEGIN (Main,2)
362
363     // USER CODE END
364
365     MAIN_vInit();
366
367     // USER CODE BEGIN (Main,3)
368     while(RS232_wait);
369     // USER CODE END
370
371     while(1)
372     {
373
374         // USER CODE BEGIN (Main,4)
375
376         // USER CODE END
377
378     }
379
380 }
381
382
383 } // End of function main

```

The screenshot shows the Infineon μVision4 IDE interface. The title bar reads "XE164 - μVision4". The menu bar includes File, Edit, View, Project, Flash, Debug, Peripherals, Tools, SVCS, Window, and Help. The toolbar has various icons for file operations. The left sidebar is the Project Explorer showing "Target 1" with files like Start\_V3.A66, MAIN.C, SCS.C, IO.C, UOC0.C, USIC0.C, CC2.C, and User Files. The main area is the code editor titled "MAIN.C\*", displaying the provided C code. At the bottom, there's a "Build Output" window showing compilation statistics: "Program Size: data=1098(near=1098) const=24(near=16) code=2426" and "XE164" - 0 Error(s), 6 Warning(s). A "Simulation" tab is also visible.

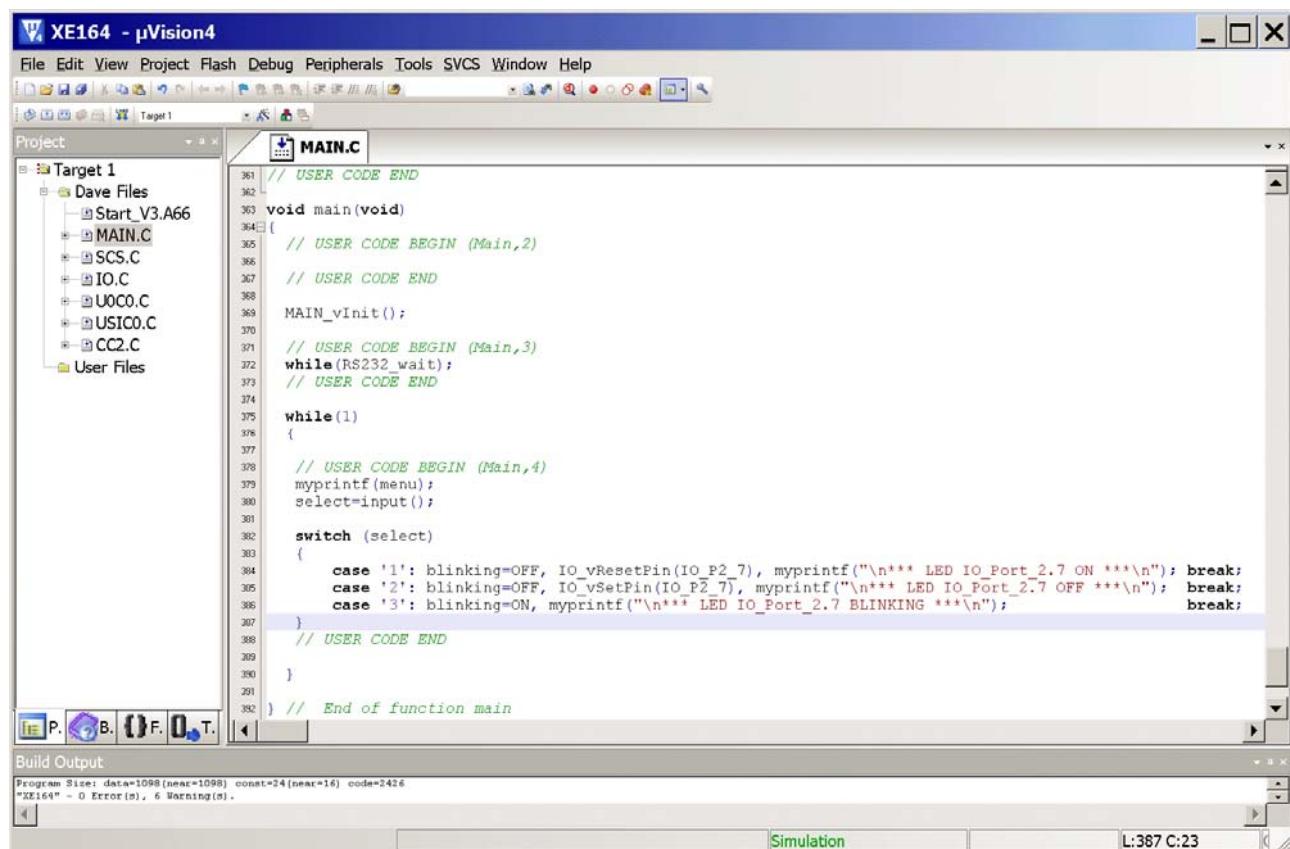
Double click MAIN.C and insert the following code in the main function into the while(1) loop:

```

myprintf(menu);
select=input();

switch (select)
{
    case '1': blinking=OFF, IO_vResetPin(IO_P2_7), myprintf("\n*** LED IO_Port_2.7 ON
***\n"); break;
    case '2': blinking=OFF, IO_vSetPin(IO_P2_7), myprintf("\n*** LED IO_Port_2.7 OFF
***\n"); break;
    case '3': blinking=ON, myprintf("\n*** LED IO_Port_2.7 BLINKING ***\n");
break;
}

```





Additional information: **Port Output Modification Register** (Source: User's Manual):

### Pn\_OMRL (n=6-11)

**Port n Output Modification Register LowXSFR ( $E9C0_H + 4 * n$ ) Reset Value:  $XXXX_H$**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PC 7	PC 6	PC 5	PC 4	PC 3	PC 2	PC 1	PC 0	PS 7	PS 6	PS 5	PS 4	PS 3	PS 2	PS 1	PS 0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Field	Bits	Type	Description
PSx (x = 0-7)	x	w	<b>Port Set Bit x</b> Setting this bit sets or toggles the corresponding bit in the port output register Pn_OUT (see <a href="#">Table 7-4</a> ). On a read access, this bit returns 0.
PCx (x = 0-7)	x + 8	w	<b>Port Clear Bit x</b> Setting this bit clears or toggles the corresponding bit in the port output register Pn_OUT. (see <a href="#">Table 7-4</a> ). On a read access, this bit returns 0.

### Function of the PCx and PSx bit fields

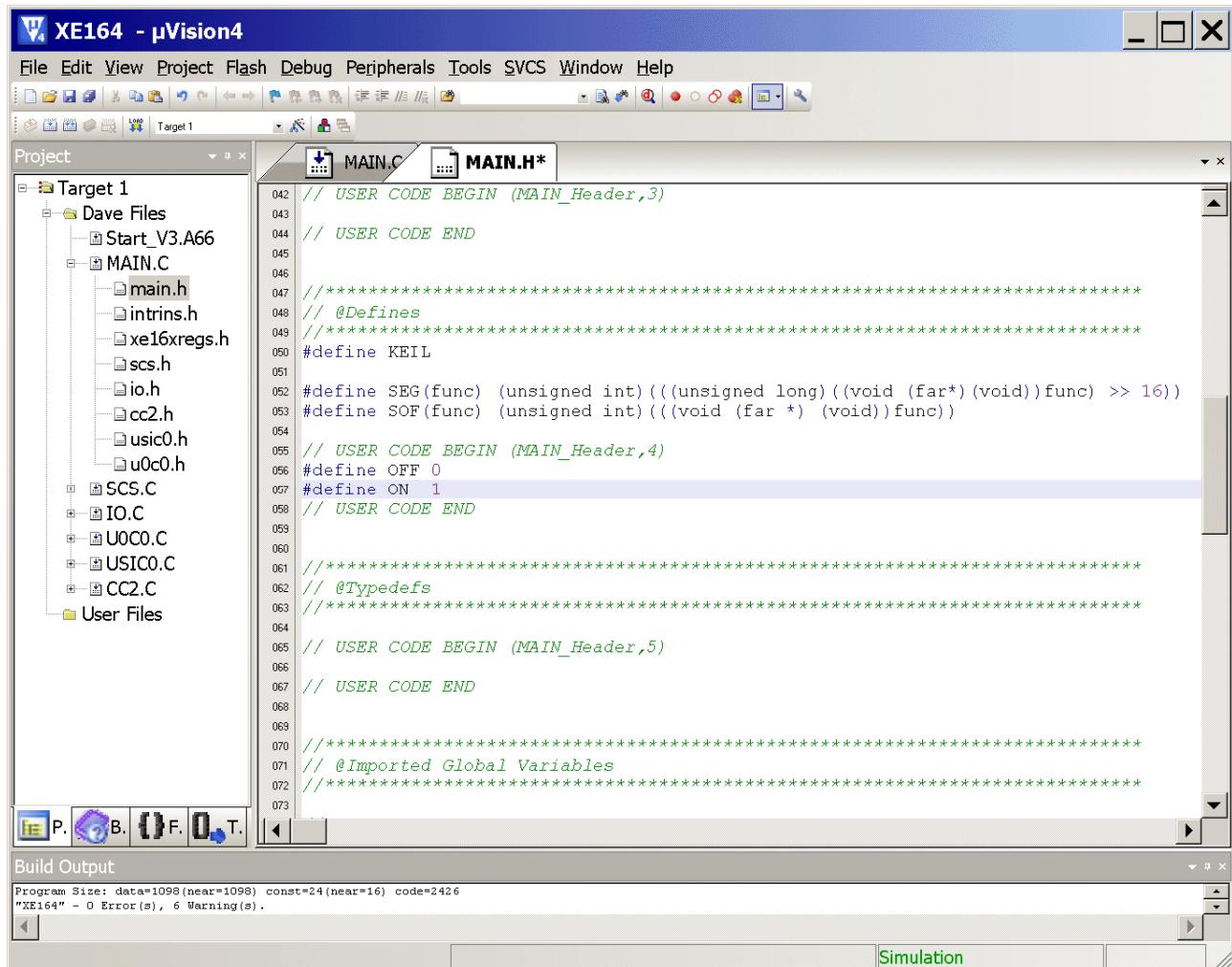
**Table 7-4 Function of the Bits PCx and PSx**

PCx	PSx	Function
0 or no write access	0 or no write access	Bit Pn_OUT.Px is not changed.
0 or no write access	1	Bit Pn_OUT.Px is set.
1	0 or no write access	Bit Pn_OUT.Px is cleared.
1	1	Bit Pn_OUT.Px is toggled.

*Note: If a bit position is not written (one out of two bytes not targeted by a byte write), the corresponding value is considered as 0. Toggling a bit requires one 16-bit write.*

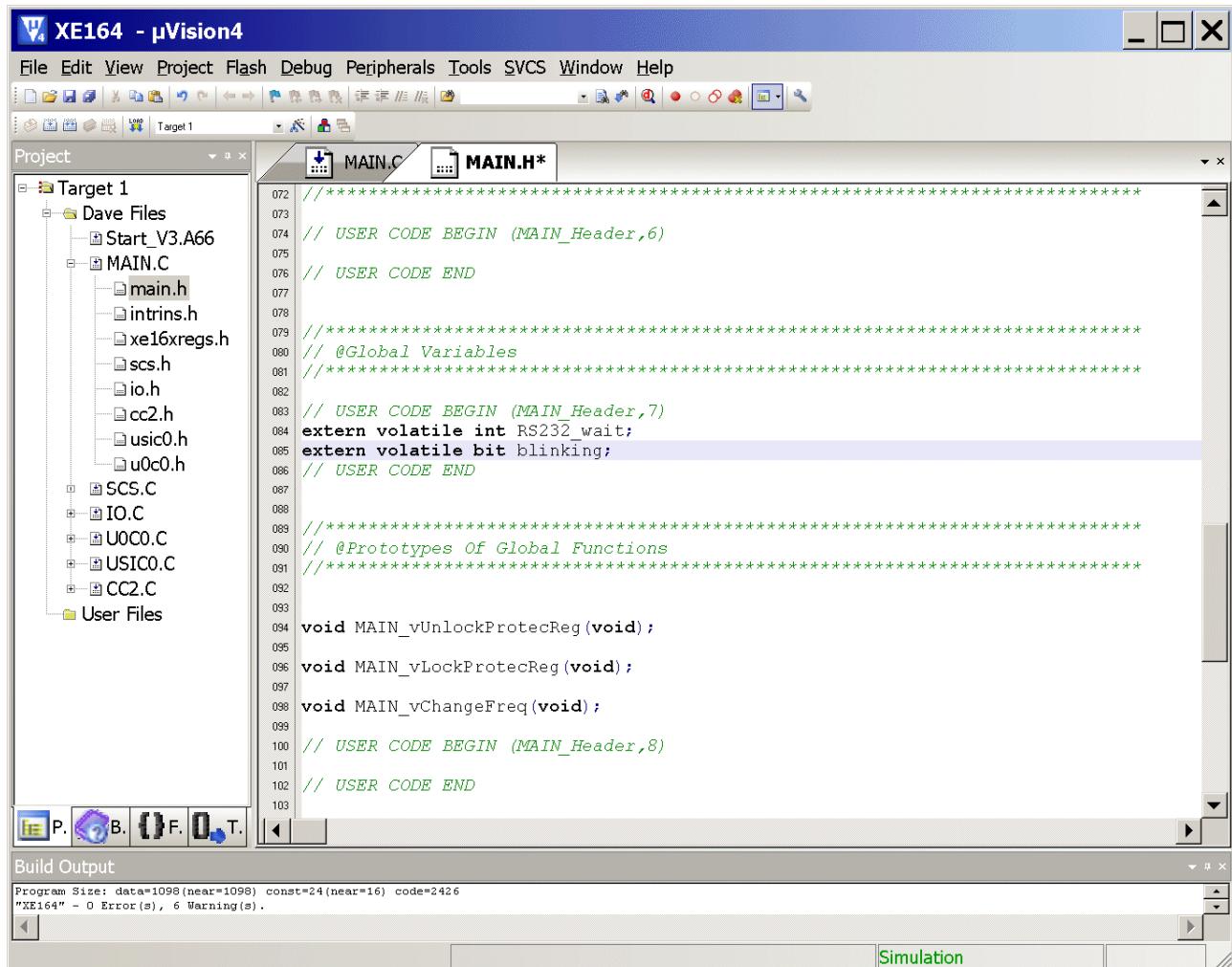
Double click Main.h and insert the following Defines:

```
#define OFF 0
#define ON 1
```



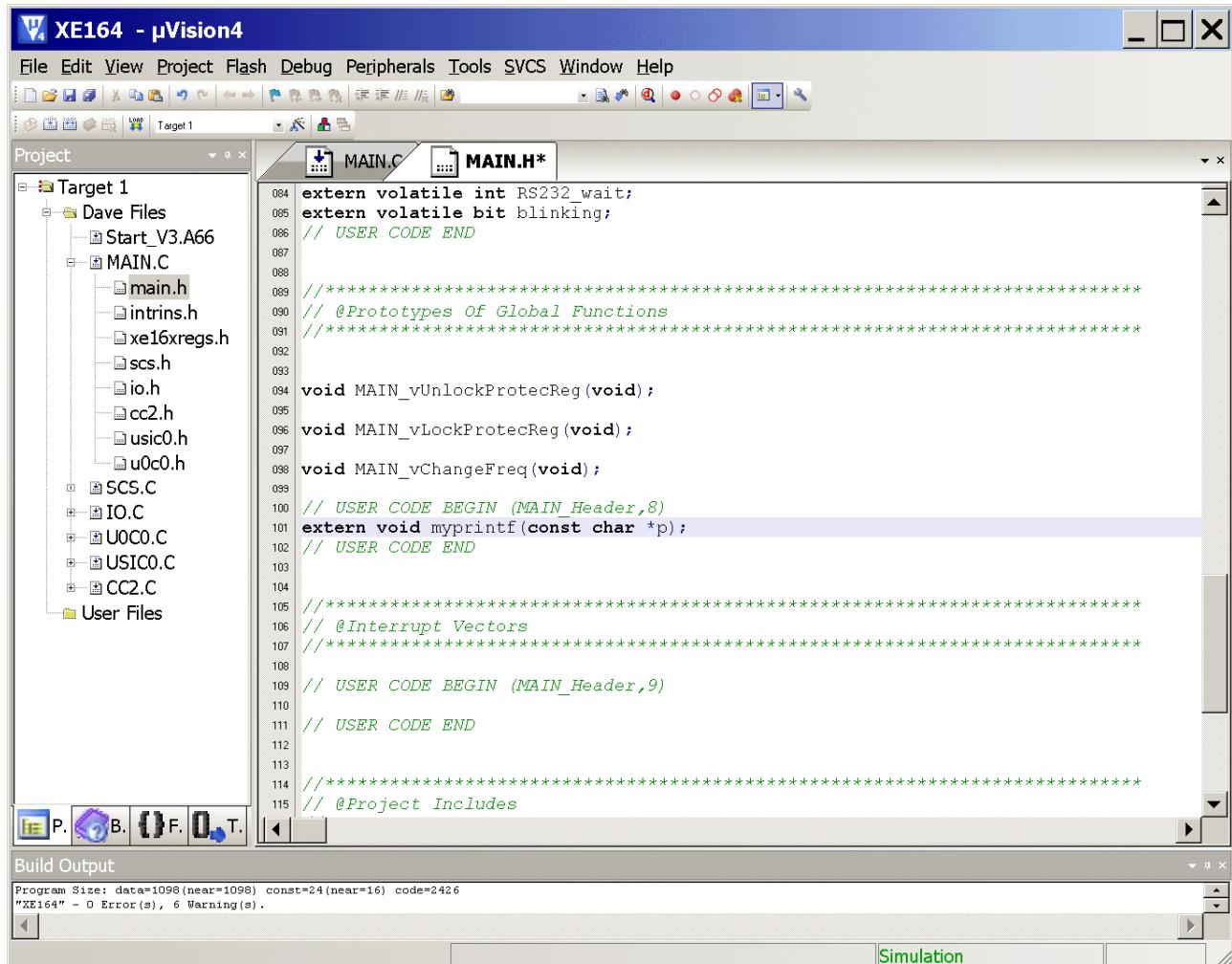
Double click Main.h and insert extern declarations “Global Variables”:

```
extern volatile int RS232_wait;
extern volatile bit blinking;
```



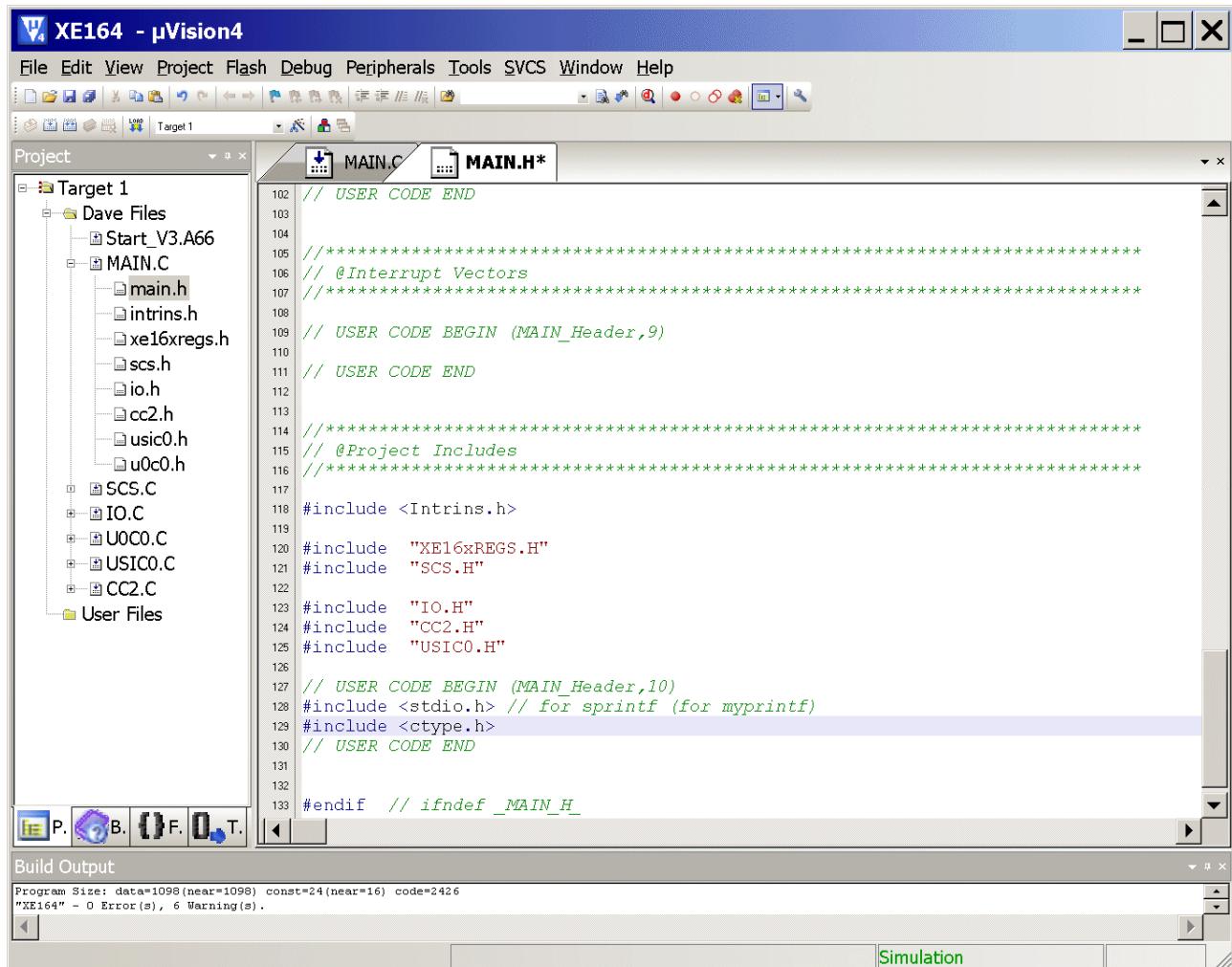
Double click Main.h and insert extern declarations “Global Functions”:

```
extern void myprintf(const char *p);
```

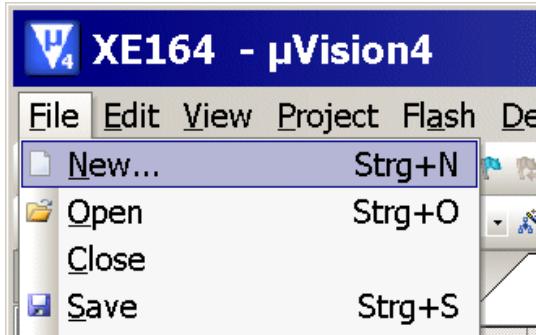


Double click Main.h and insert include files:

```
#include <stdio.h> // for sprintf (for myprintf)
#include <ctype.h>
```



File – New



Insert:

```
#include "main.h"

void myprintf(const char *p)
{
    while(*p)
    {
        U0C0_ASC_vSendData(*p++);
    }
}

/*
// Example 1 (use of myprintf):
// =====

void main(void)
{
    myprintf("Hello World!\r\n");
}

// Example 2 (use of myprintf):
// =====

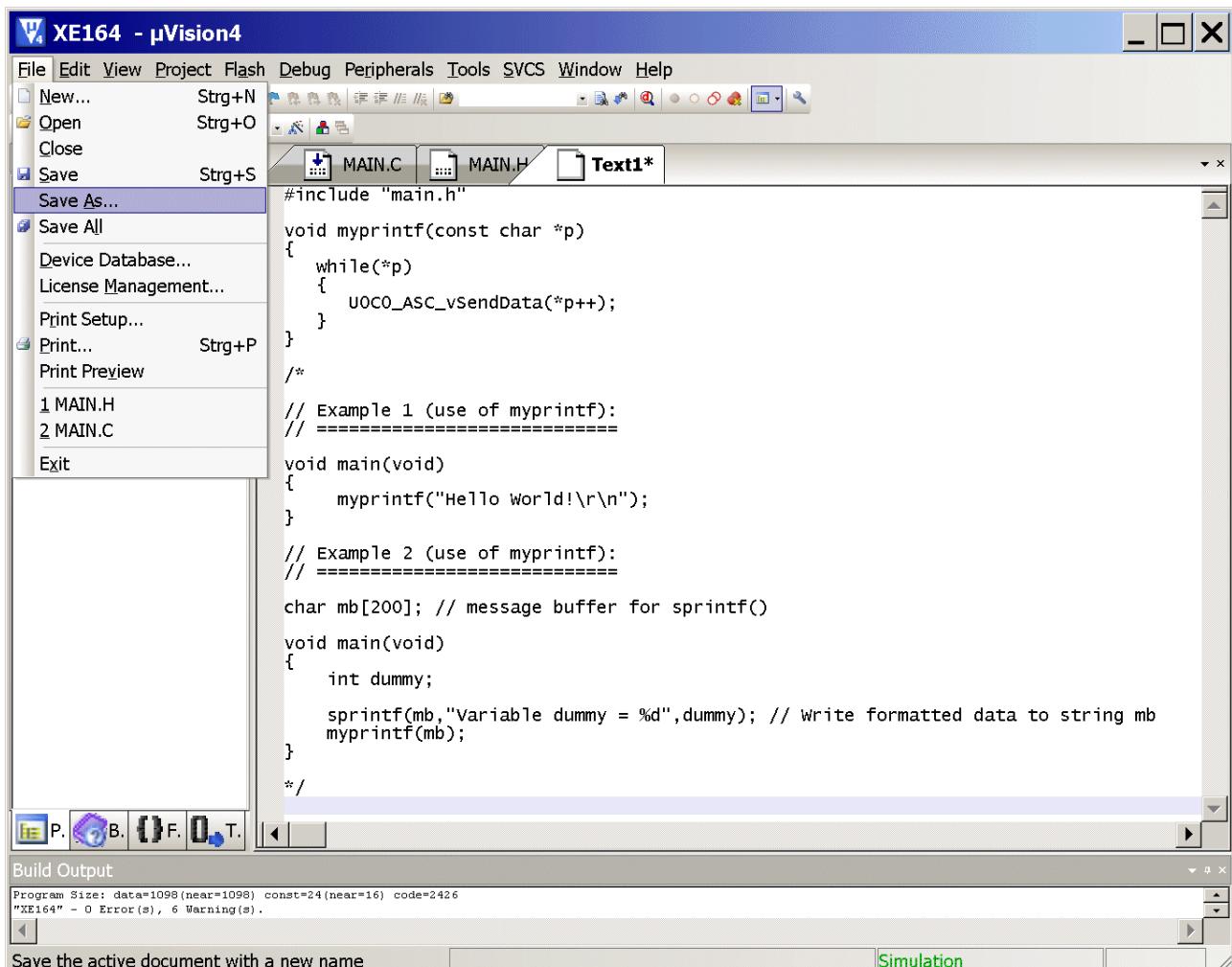
char mb[200]; // message buffer for sprintf()

void main(void)
{
    int dummy;

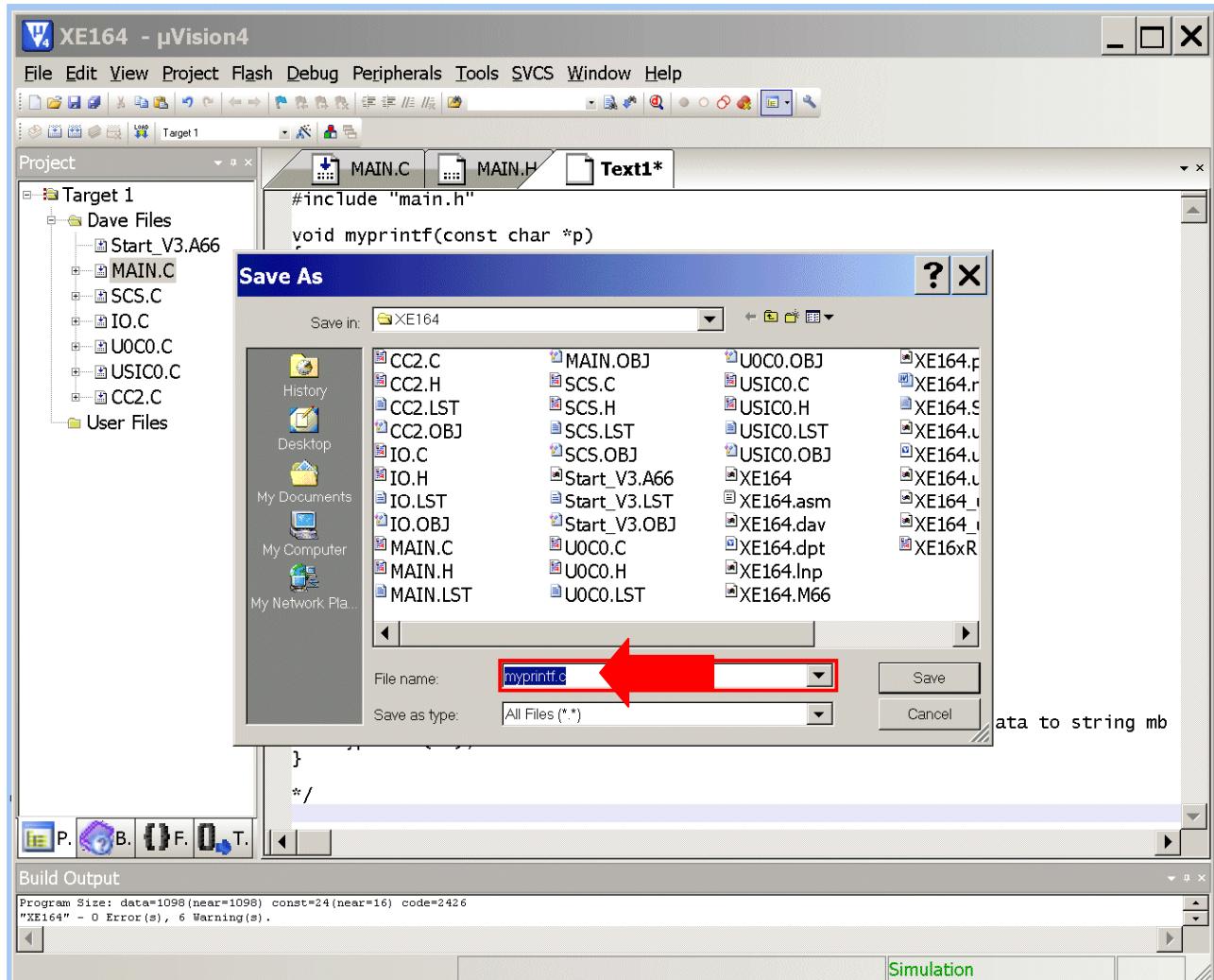
    sprintf(mb,"Variable dummy = %d",dummy); // Write formatted data to string mb
    myprintf(mb);
}

*/
```

## File – Save As...



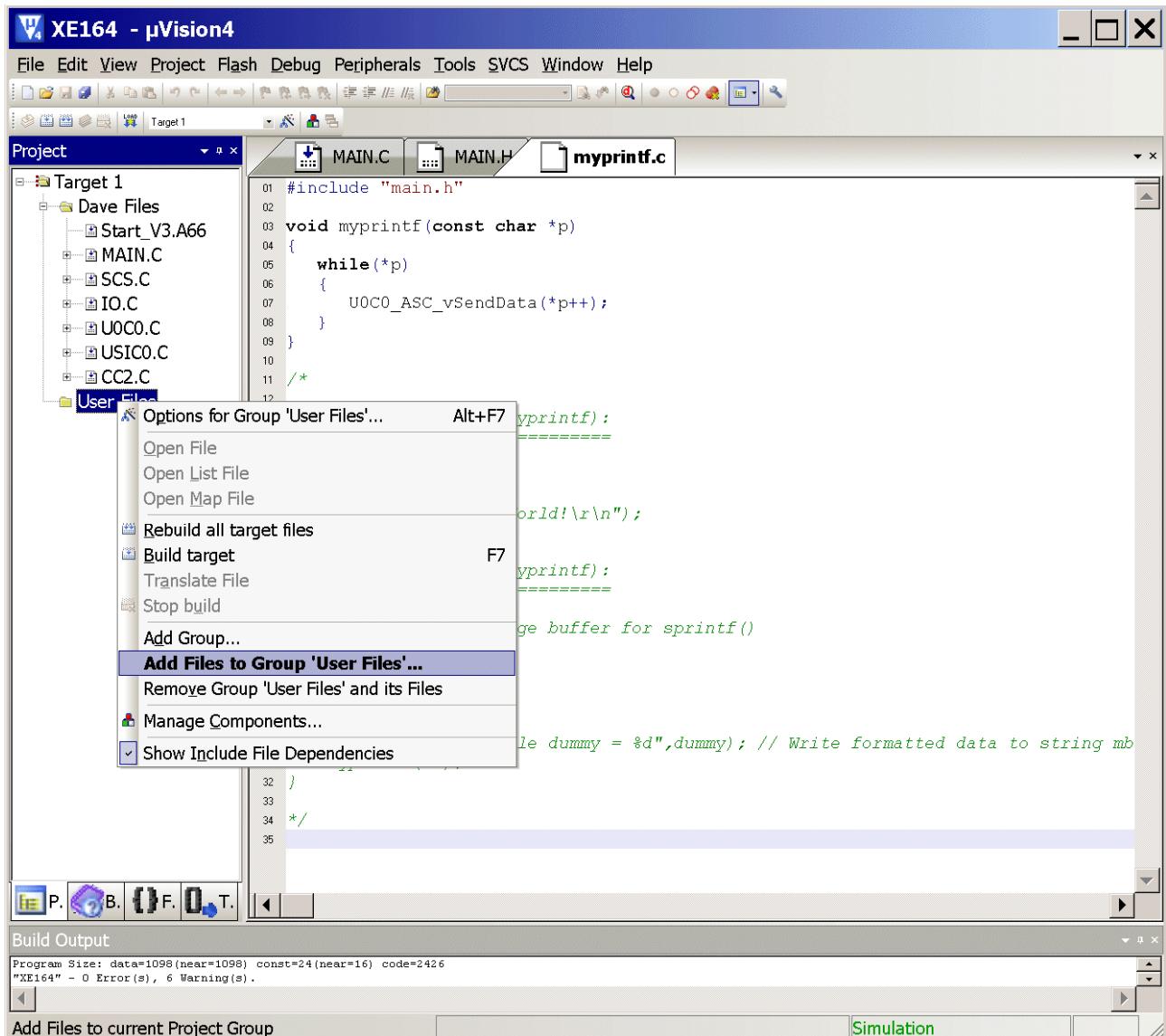
Insert: myprintf.c



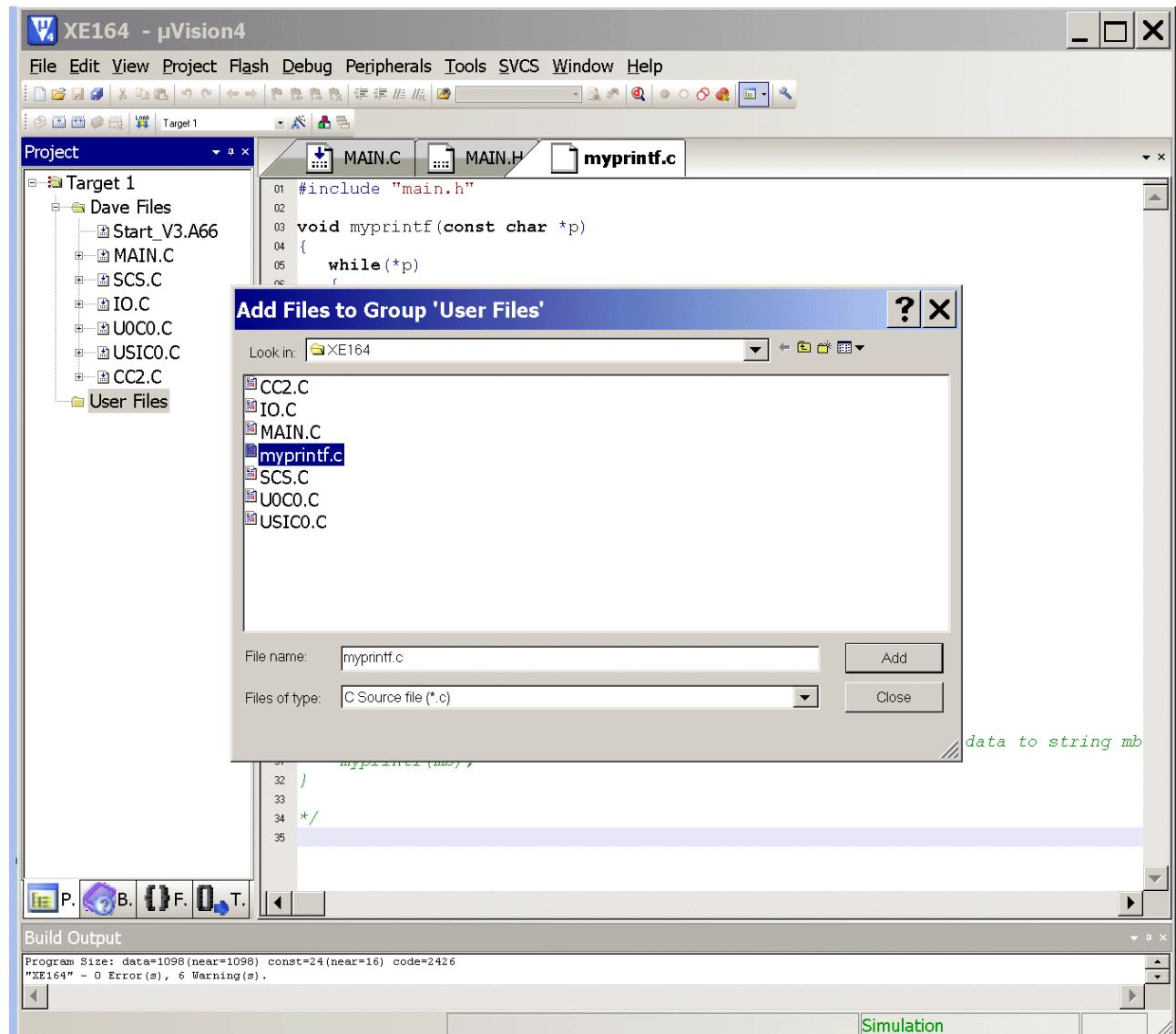
Save



Mouse position: Project Window, User Files: click right mouse button  
**click Add Files to Group 'User Files'**



Click/select myprintf.c



Add  
Close

**XE164 - µVision4**

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

Project Target 1

MAIN.C MAIN.H myprintf.c

```

01 #include "main.h"
02
03 void myprintf(const char *p)
04 {
05     while(*p)
06     {
07         U0C0_ASC_vSendData(*p++);
08     }
09 }
10 /*
11
12 myprintf.c
13 =====
14
15 void main(void)
16 {
17     myprintf("Hello World!\r\n");
18 }
19
20 // Example 2 (use of myprintf):
21 // =====
22
23 char mb[200]; // message buffer for sprintf()
24
25 void main(void)
26 {
27     int dummy;
28
29     sprintf(mb,"Variable dummy = %d",dummy); // Write formatted data to string mb
30     myprintf(mb);
31 }
32
33 */
34
35

```

P B F T

Build Output

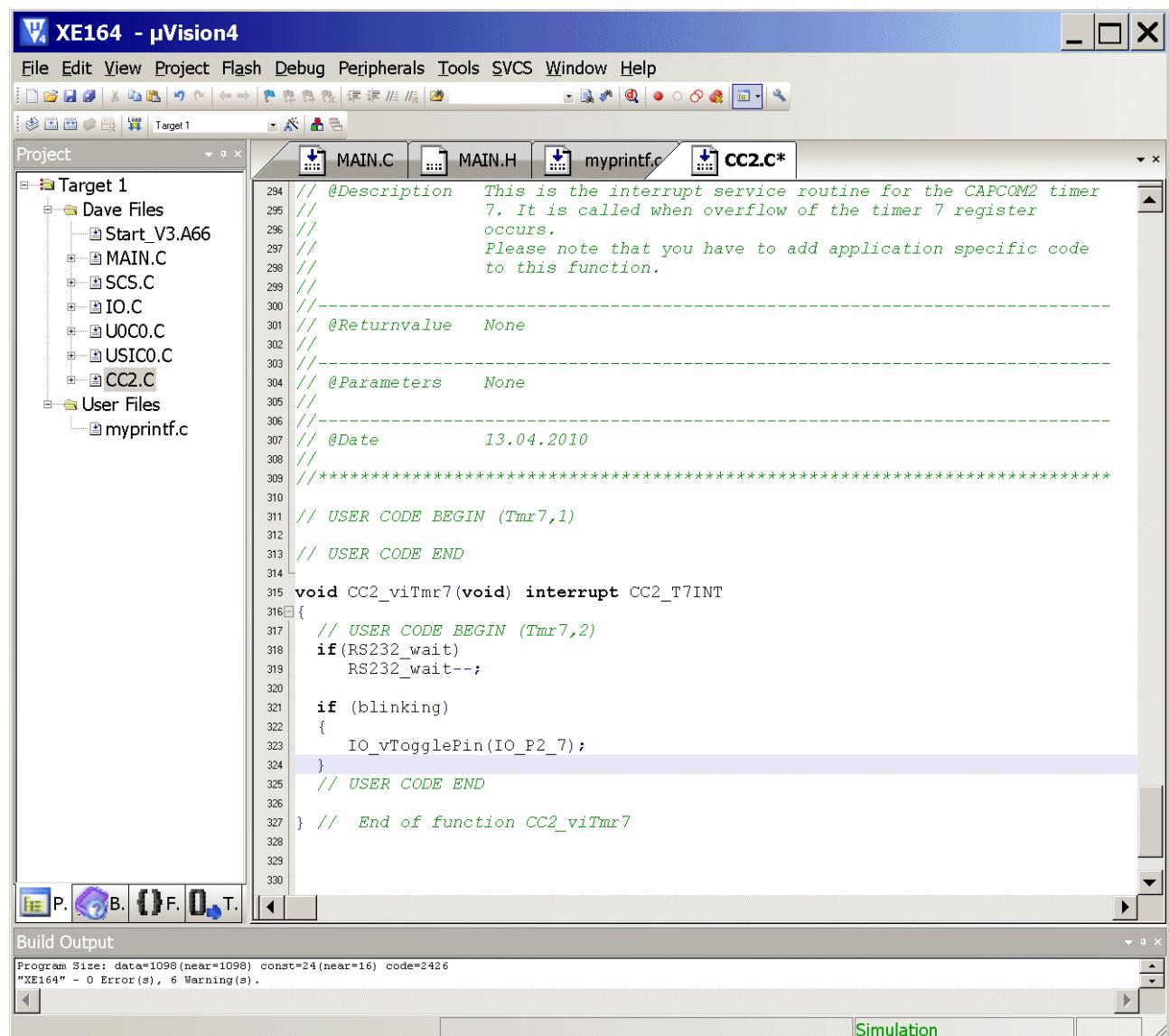
Program Size: data=1098(near=1098) const=24(near=16) code=2426  
"XE164" - 0 Error(s), 6 Warning(s).

Simulation

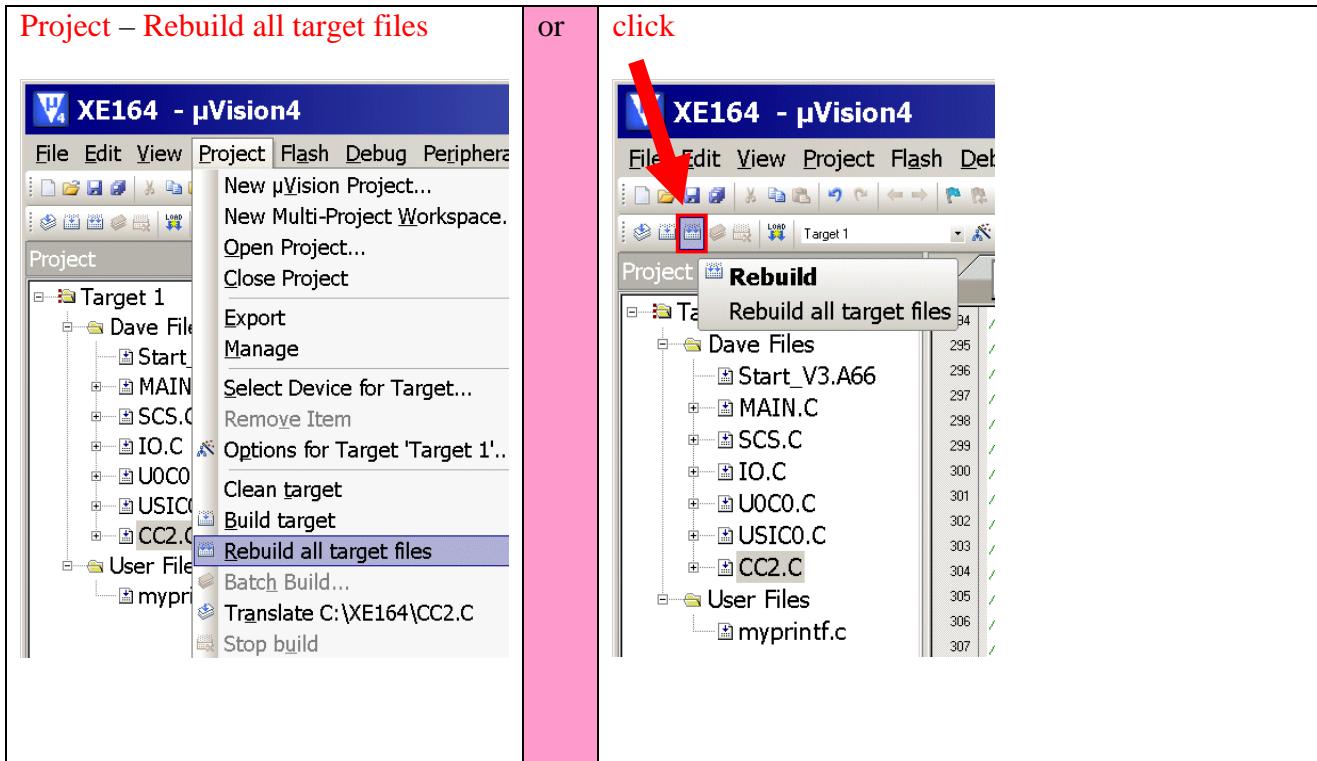
Double click CC2.C insert Code (CAPCOM 2 Timer 7 Interrupt Service Routine):

```
if(RS232_wait)
    RS232_wait--;

if (blinking)
{
    IO_vTogglePin(IO_P2_7);
}
```



Generate your application program:



**XE164 - µVision4**

**File Edit View Project Flash Debug Peripherals Tools SvCS Window Help**

**Target 1**      **Dave Files**      **Start\_V3\_A66**

- MAIN.C
- SCS.C
- IO.C
- UOCO.C
- USICO.C
- CC2.C

**User Files**

- myprintf.c

```

372     while(RS232_wait());
373     // USER CODE BEGIN (Main,4)
374     while(1)
375     {
376         // USER CODE BEGIN (Main,4)
377         myprintf("Main\r\n");
378         myprintf("Main\r\n");
379         myprintf("Main\r\n");
380         selectInput();
381
382         switch (select)
383         {
384             case '1': blinking=OFF, IO_ResetPin(IO_P2_7), myprintf("\n*** LED IO_Port_2_7 ON ***\n");
385             case '2': blinking=OFF, IO_SetPin(IO_P2_7), myprintf("\n*** LED IO_Port_2_7 OFF ***\n");
386             case '3': blinking=ON, myprintf("\n*** LED IO_Port_2_7 BLINKING ***\n");
387         }
388         // USER CODE END
389     }
390 }
391
392 } // End of function main
393
394

```

**Build Output**

```

Build target 'Target 1'

assembling Start_V3_A66...
compiling MAIN.C...
compiling SCS.C...
SCS.C(582): warning C174: 'Scs_IdelayByTimer': unreferenced 'static' function
SCS.C(551): warning C174: 'Scs_IdelayByNops': unreferenced 'static' function
compiling IO.C...
IO.C(376): warning C174: 'Scs_EnableHighPreosc': unreferenced 'static' function
compiling UOCO.C...
UOCO.C(376): warning C174: 'Scs_EnableHighPreosc': unreferenced 'static' function
compiling USICO.C...
USICO.C(376): warning C174: 'Scs_EnableHighPreosc': unreferenced 'static' function
compiling CC2.C...
CC2.C(376): warning C174: 'Scs_EnableHighPreosc': unreferenced 'static' function
compiling myprintf.c...
myprintf.c(376): warning C174: 'Scs_EnableHighPreosc': unreferenced 'static' function
linking...
Program Size: data=1101(hex=1101) const=242(hex=212) code=2514
creating hex file from "XE164...".
"XE164" - 0 Error(s), 7 Warning(s).

```

**Simulation**

L:392 C:5

**Note:**

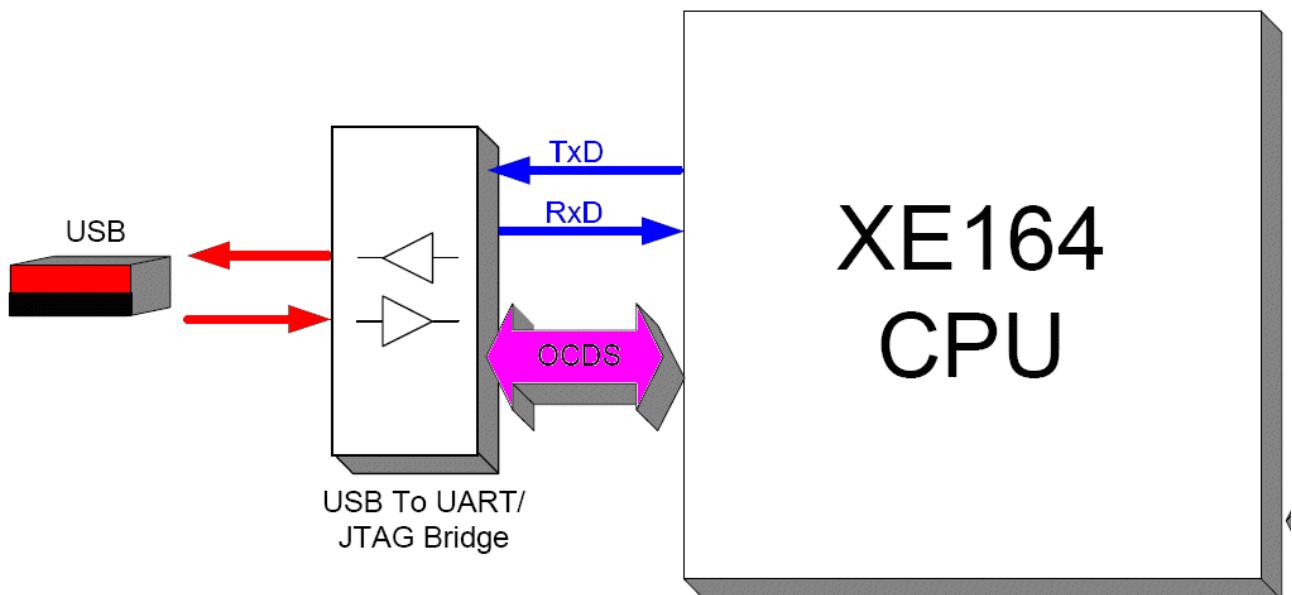
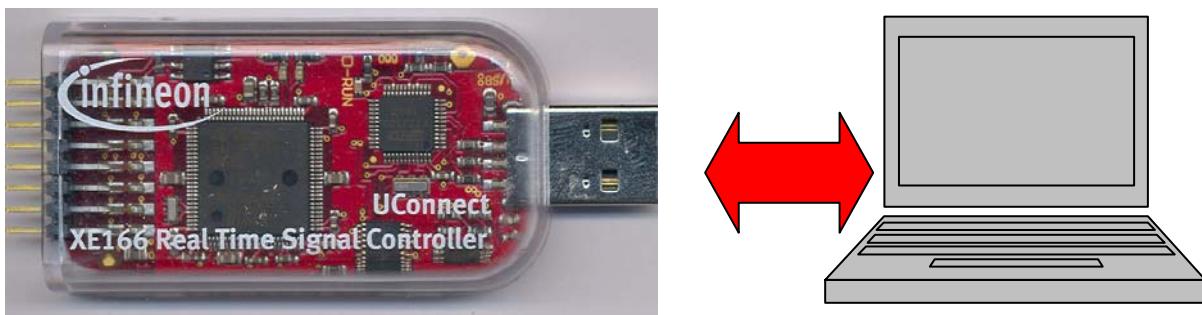
Programming is now complete.

Unfortunately it is not possible to test your program with the Keil Simulator because this feature is currently not supported.

Therefore we are going to **load** (On Chip Flash Programming) and **run** your program on the UConnect-CAN XE164 in the next chapter.

### 5.) Running your first programming example:

Make sure that the UConnect-CAN XE164 is still connected to the host computer:



#### USB Connection:

- .) used for: **UART communication** (the USIC0\_CH0/UART/RS232/serial interface is available via USB as a virtual COM port of the second USB channel of the FTDI FT2232 Dual USB to UART/JTAG interface).
- .) used for: **On-Chip-Flash-Programming and Debugging** (first USB channel of the FTDI FT2232 Dual USB to UART/JTAG interface).
- .) the USB connection works also as the power supply.

**Note:**

Now we need a terminal program which is able to handle a virtual COM port (COM12)!

As an example of “any terminal program” we are going to use Docklight.

Docklight can be downloaded @ <http://www.docklight.de> :

**Docklight - Download - Infineon Technologies**

File Edit View Favorites Tools Help

Back Search Folders Favorites

Address http://www.docklight.de/download\_en.htm Go Links

**Software Archive**

**Latest Releases**

<a href="#">Docklight V1.7</a>	Download Docklight V1.7.37 for Windows Vista, Windows XP, Windows 2000, Windows NT, Windows ME, Windows 98 (3143 KB)
<a href="#">Docklight Scripting V1.7</a>	Download Docklight Scripting V1.7.37 for Windows Vista, Windows XP, Windows 2000, Windows NT, Windows ME, Windows 98 (4314 KB)

**Access to previous and obsolete versions of Docklight**

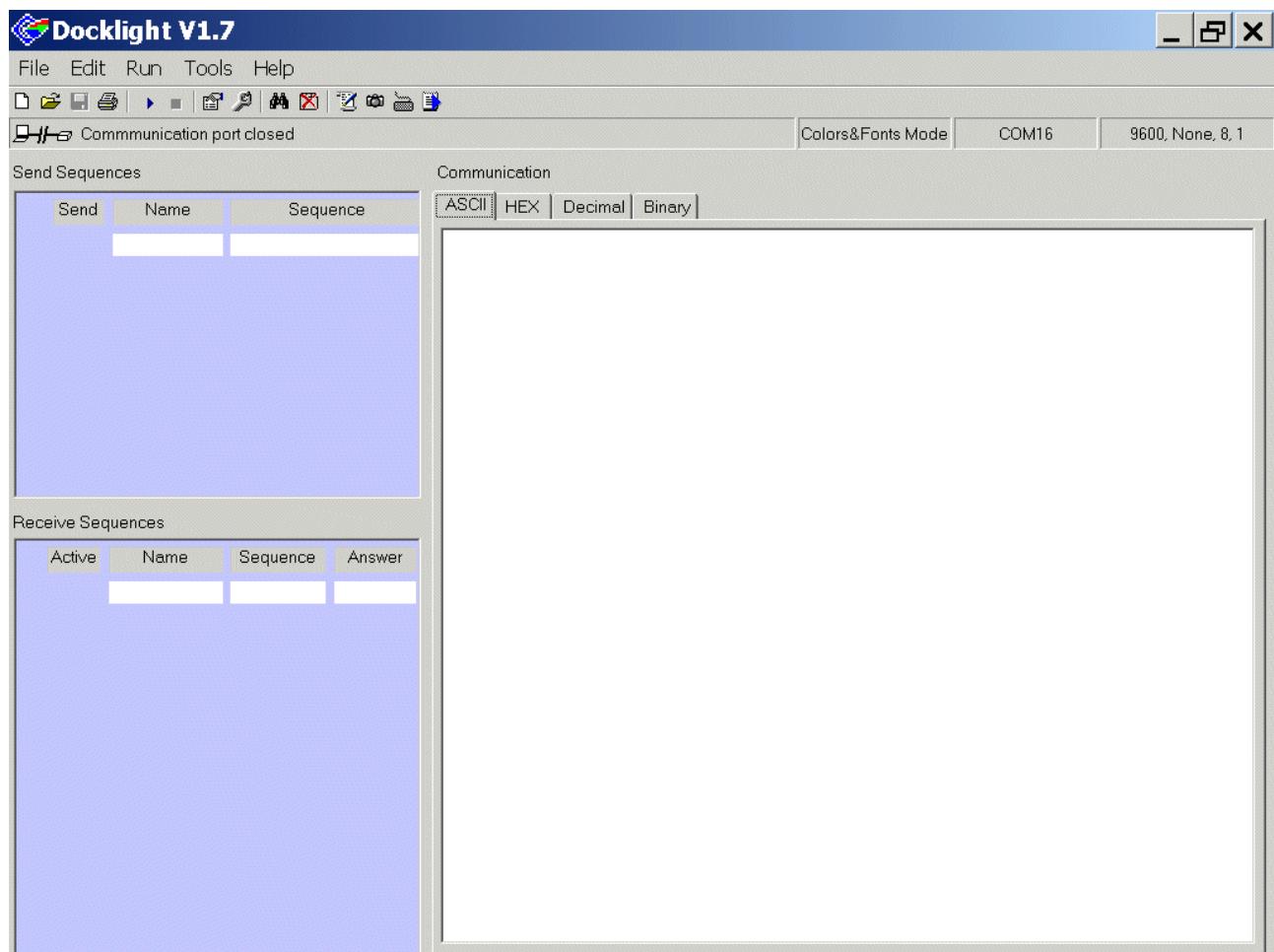
<a href="#">Docklight Scripting V1.6</a>	Download Docklight Scripting V1.6 (3960 KB, Release 01/2007)
<a href="#">Docklight V1.6</a>	Download Docklight V1.6 (3027 KB, Release 01/2007)
<a href="#">Docklight Scripting V1.5</a>	Download Docklight Scripting V1.5 (3909 KB, Release 09/2004)
<a href="#">Docklight Scripting V1.4</a>	Download Docklight Scripting V1.4 (3868 KB, Release 05/2004)
<a href="#">Docklight V1.4</a>	Download Docklight V1.4 (3028 KB, Release 05/2004)
<a href="#">Docklight V1.3</a>	Download Docklight V1.3 (3048 KB, Release 01/2004)
<a href="#">Docklight V1.2</a>	Download Docklight V1.2 (2866 KB, Release 02/2003)
<a href="#">Docklight V1.1</a>	Download Docklight V1.1 (2850 KB, Release 09/2002)
<a href="#">Docklight V1.0</a>	Download Docklight V1.0 (2834 KB, Release 04/2002)

[← back to the Docklight main page](#)

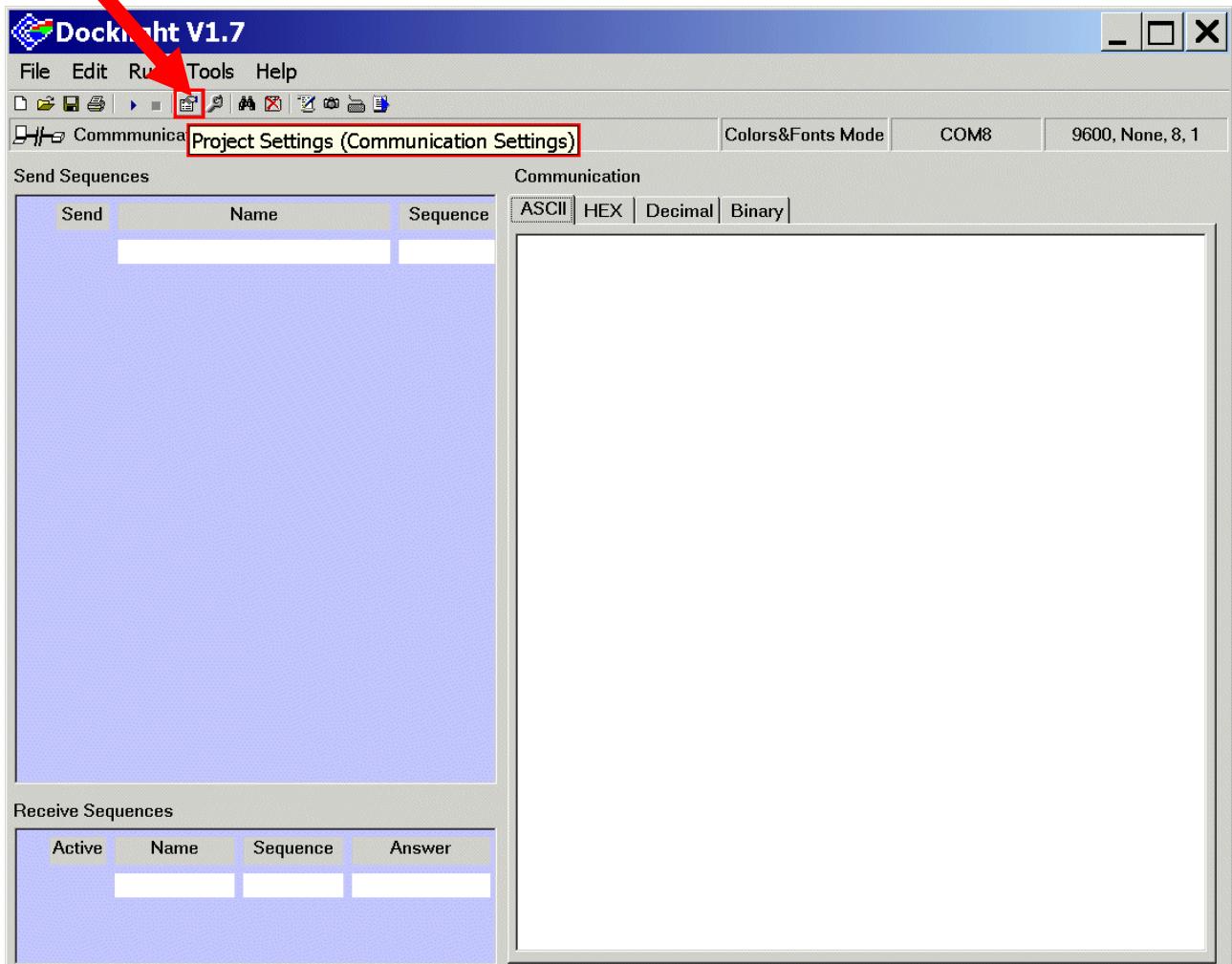
Internet



Now, start Docklight:



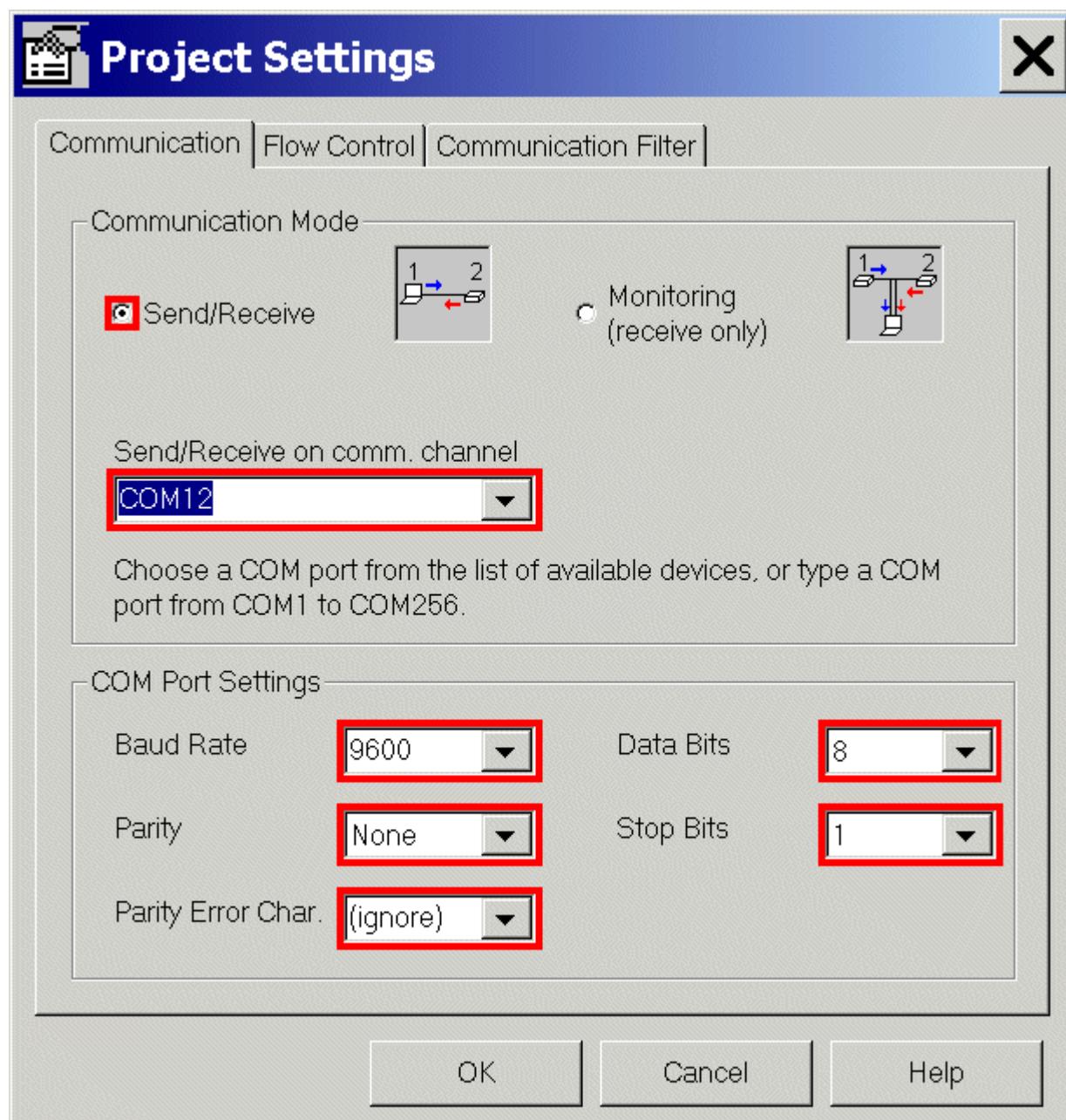
Click: Project Settings



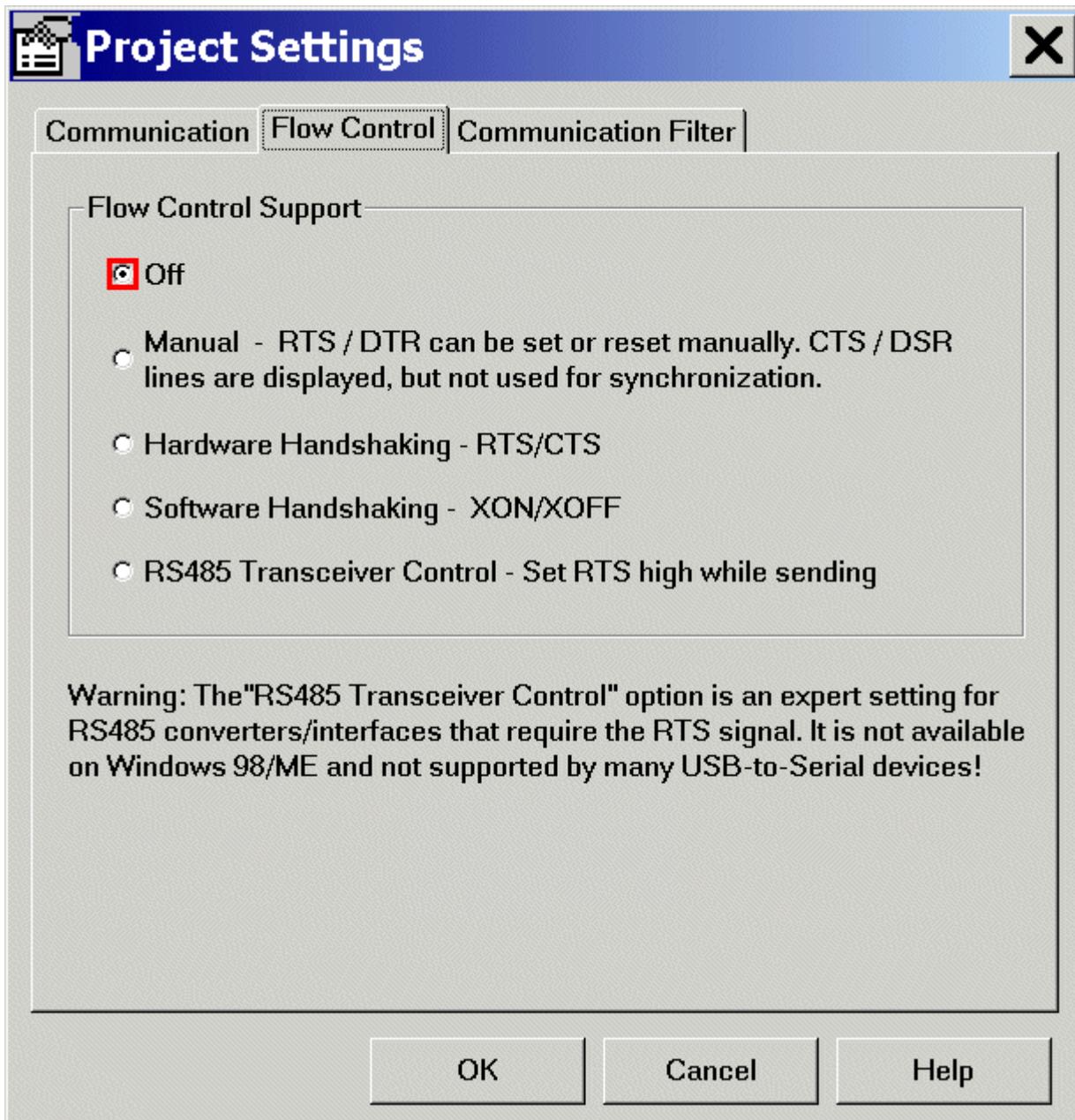
## Project Settings:

Communication: Communication Mode: **click/check**  Send/Receive

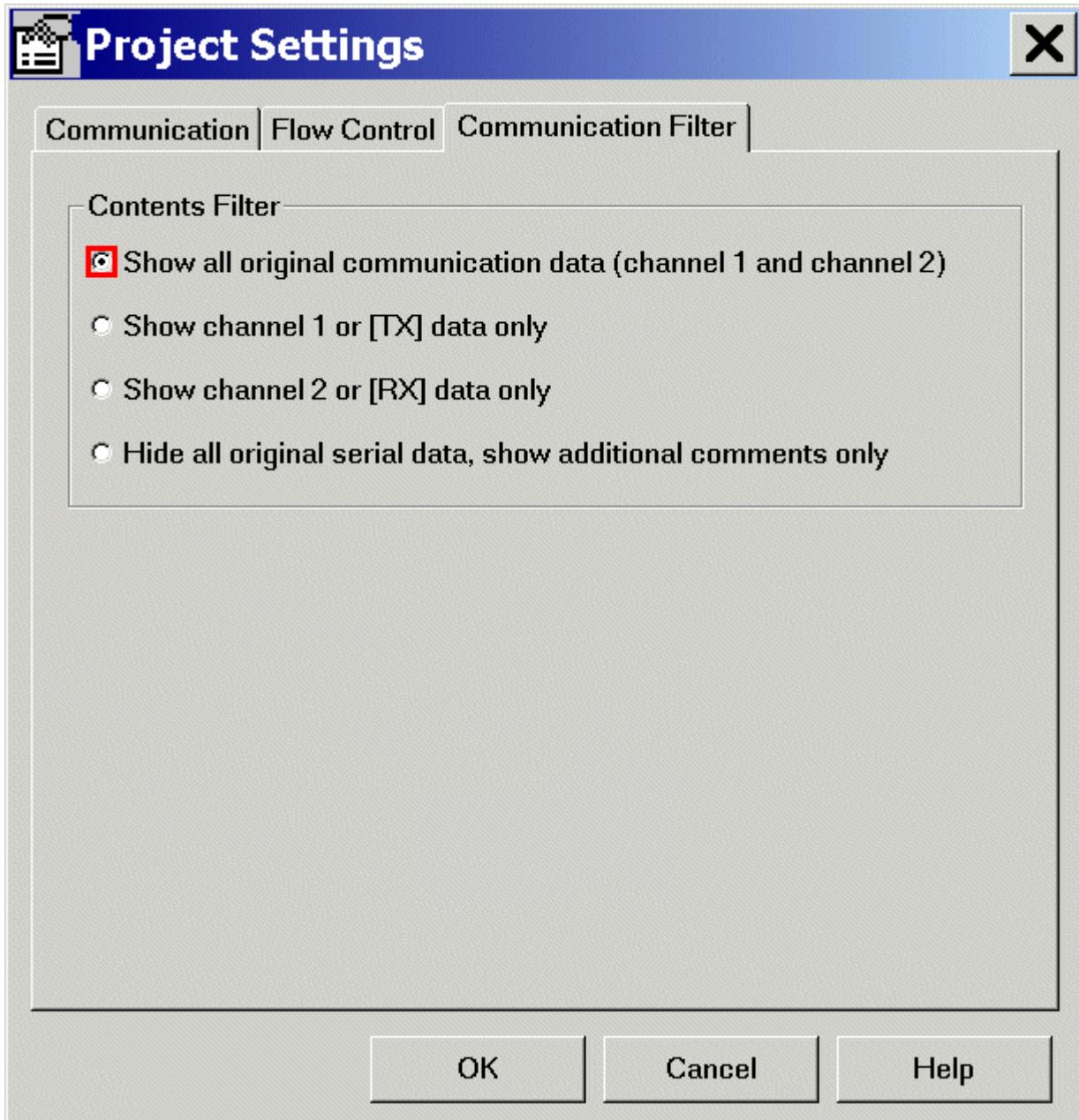
## Project Settings:

Communication: Communication Mode: Send/Receive on comm. channel: **select** COM12Project Settings: Communication: COM Port Settings: Baud Rate: **select** 9600Project Settings: Communication: COM Port Settings: Parity: **select** NoneProject Settings: Communication: COM Port Settings: Parity Error Char.: **select** (ignore)Project Settings: Communication: COM Port Settings: Data Bits: **select** 8Project Settings: Communication: COM Port Settings: Stop Bits: **select** 1

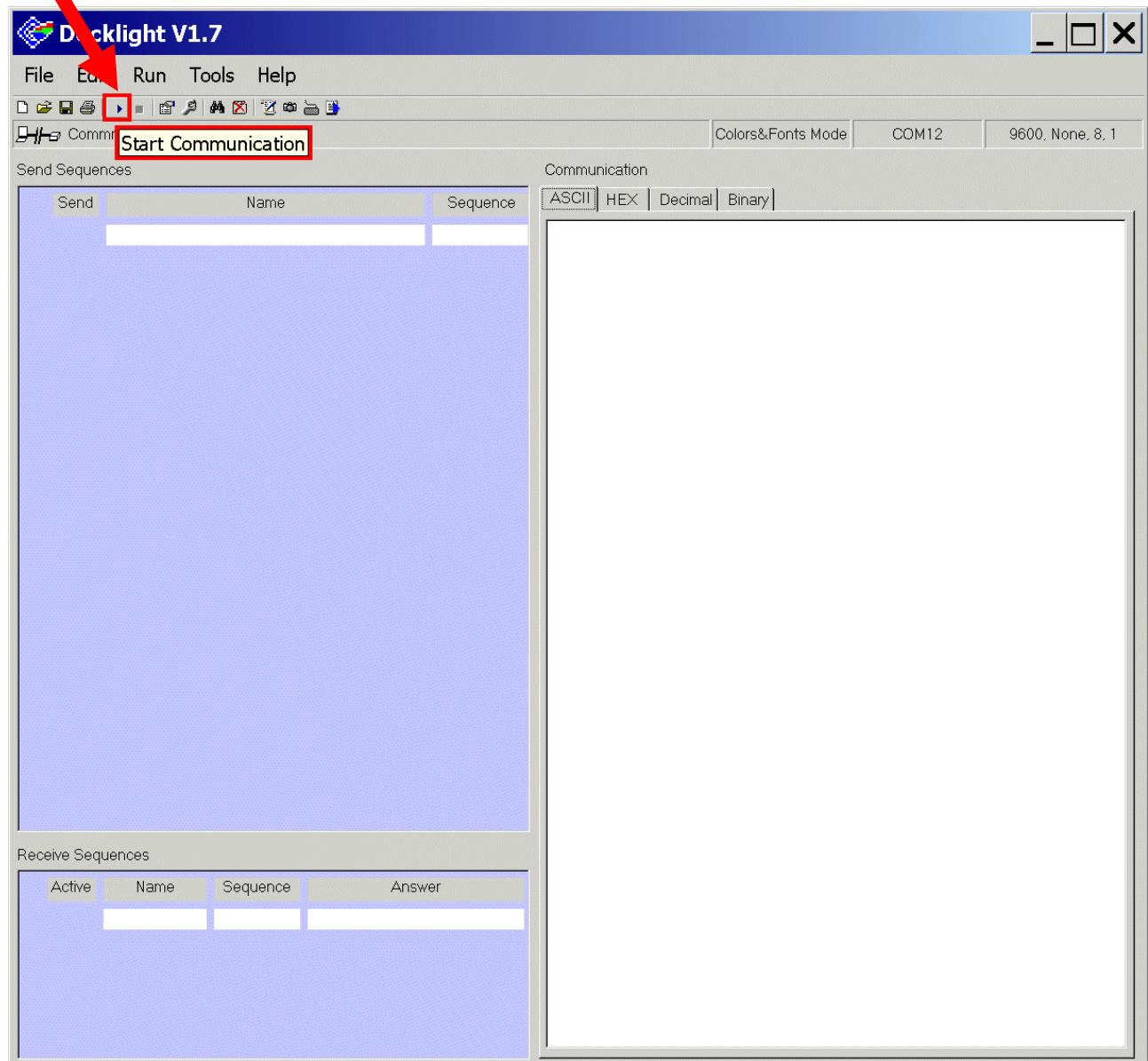
Project Settings: Flow Control: Flow Control Support: [click](#)  Off

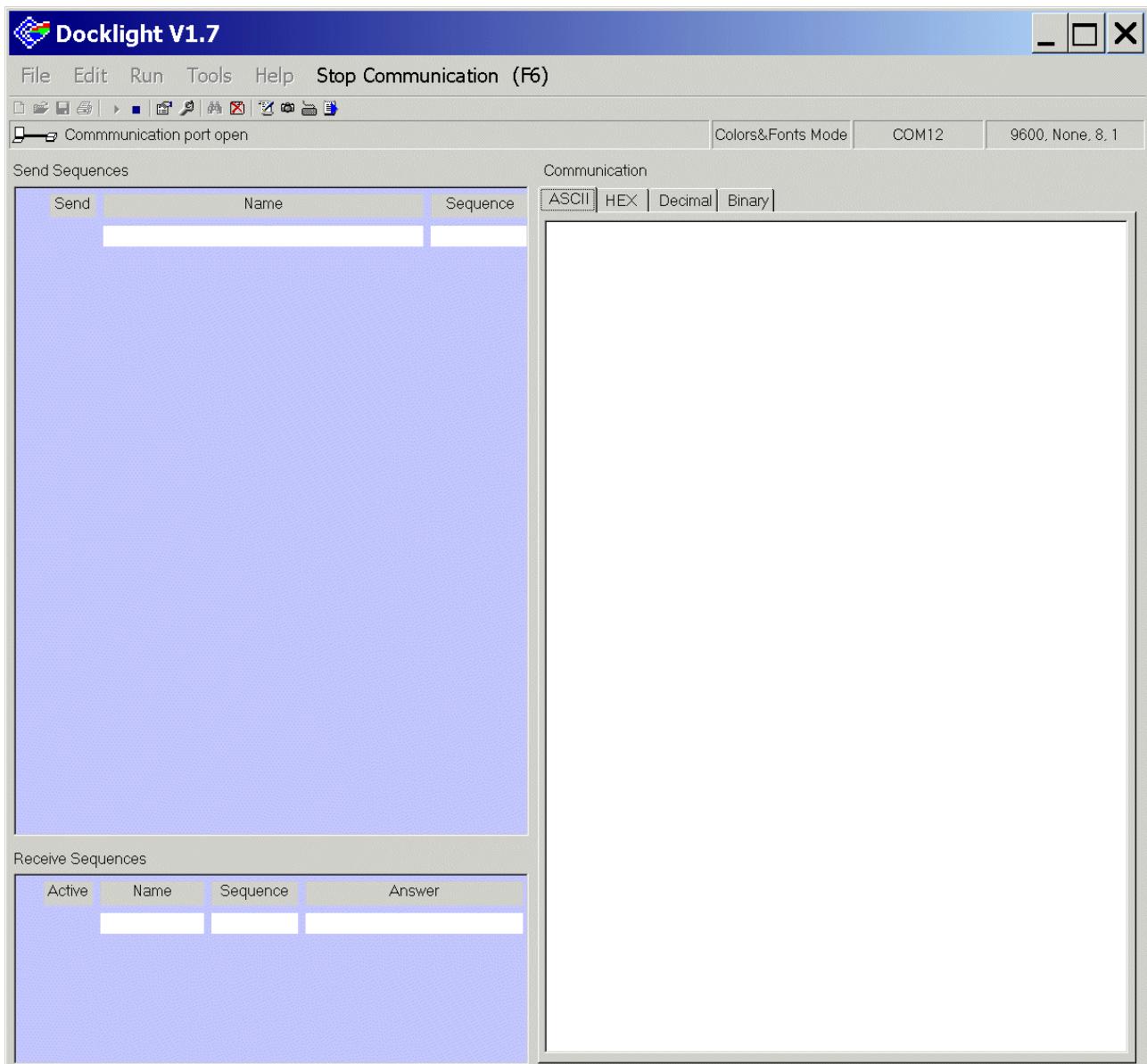


## Project Settings:

Communication Filter: Contents Filter: [click](#)  Show all original communication data**OK**

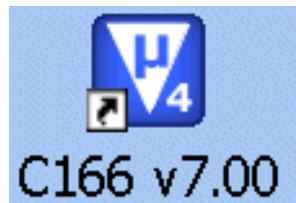
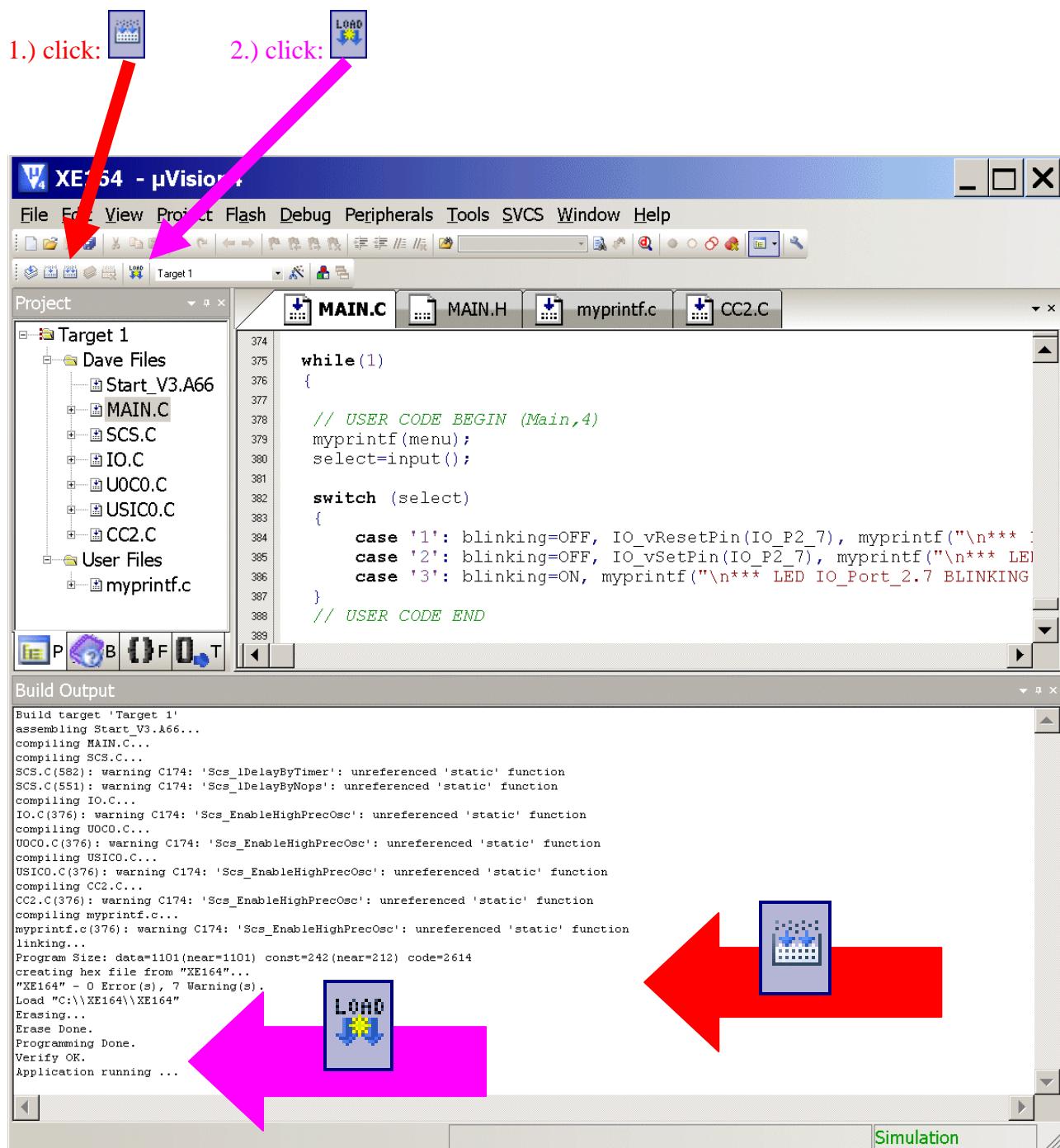
Click: 



**Note:**

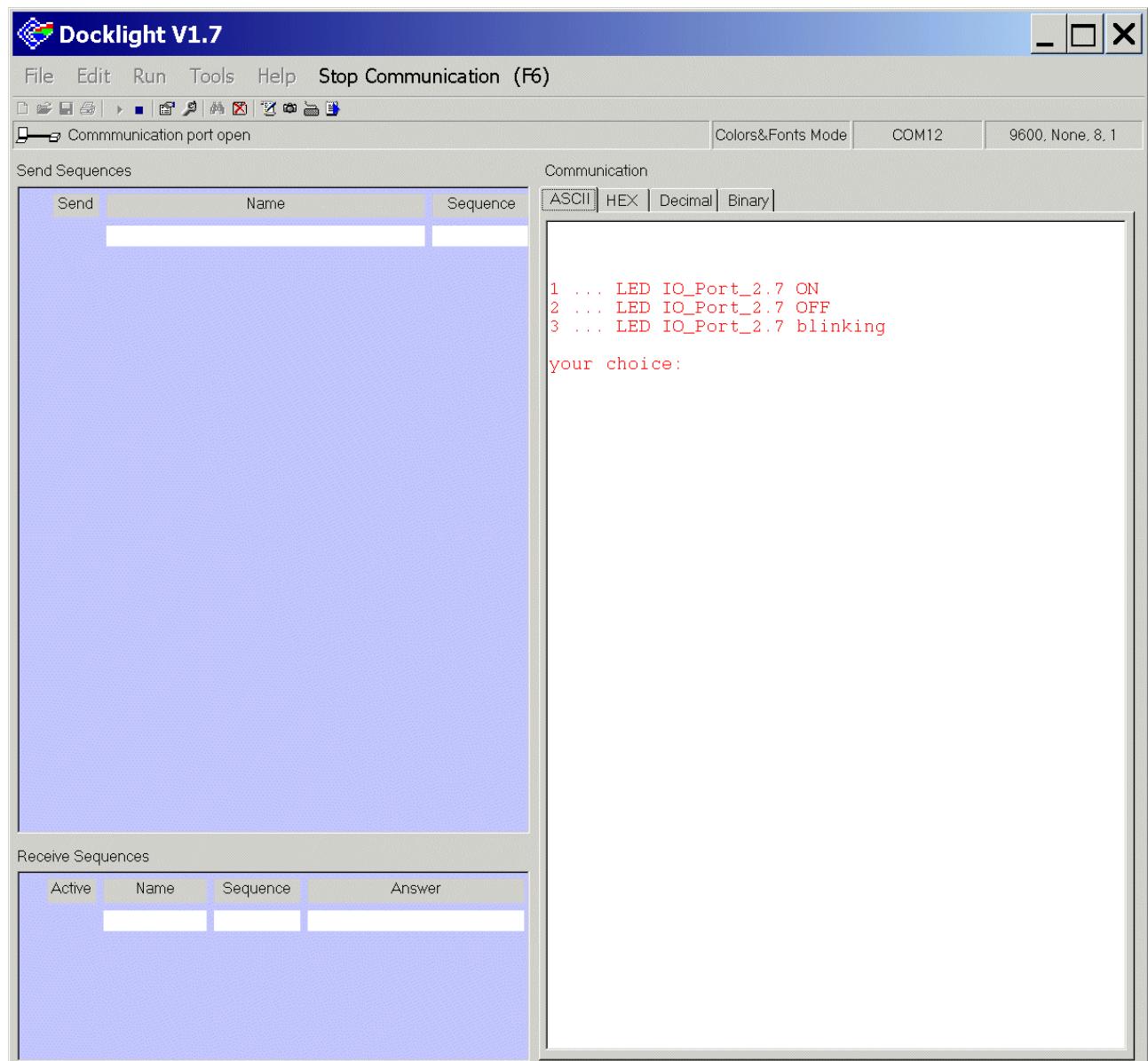
Docklight is now ready for serial communication!



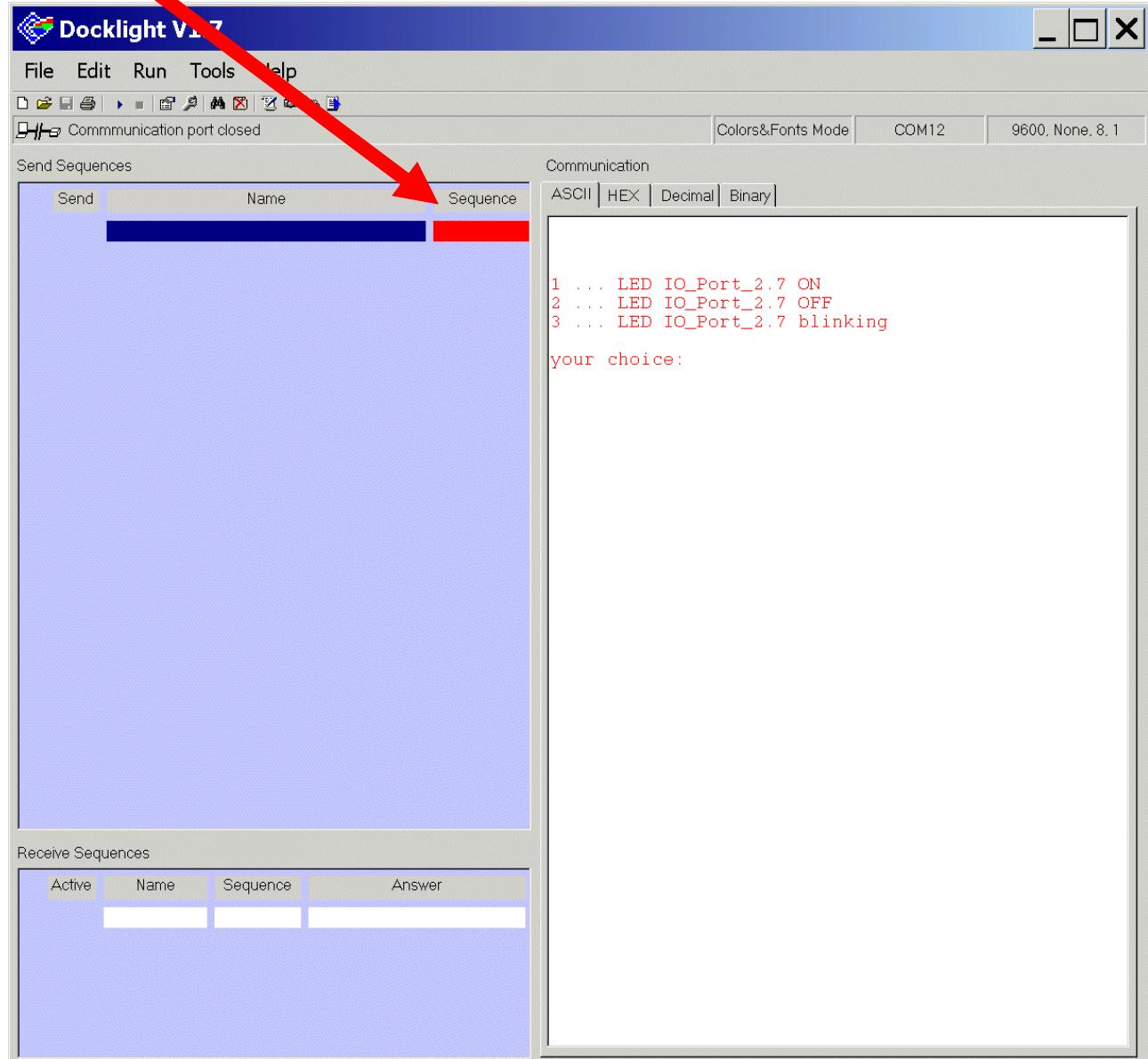

Go to μVision4:




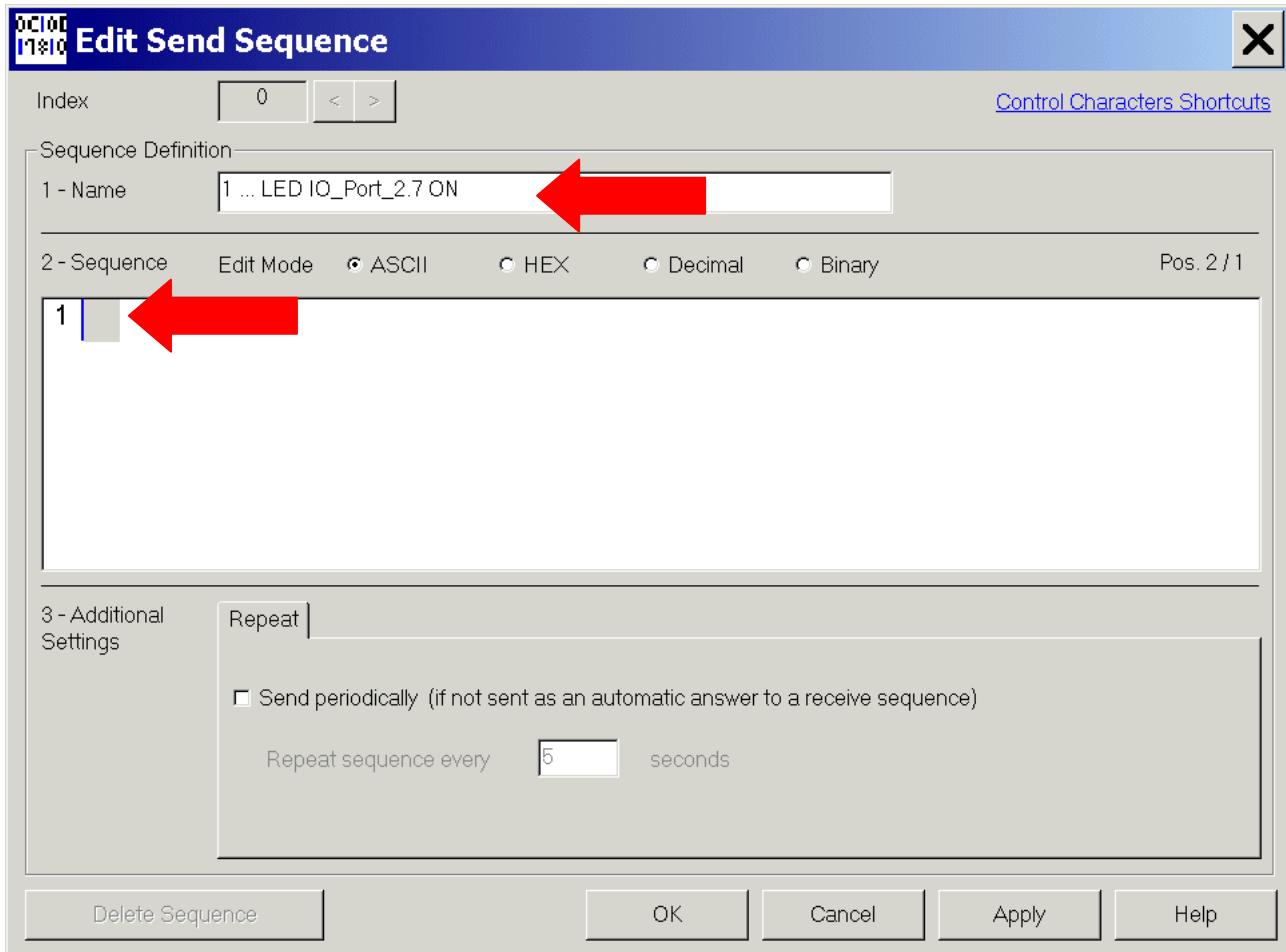
Go to Docklight and see the result:



Double click inside the red box:

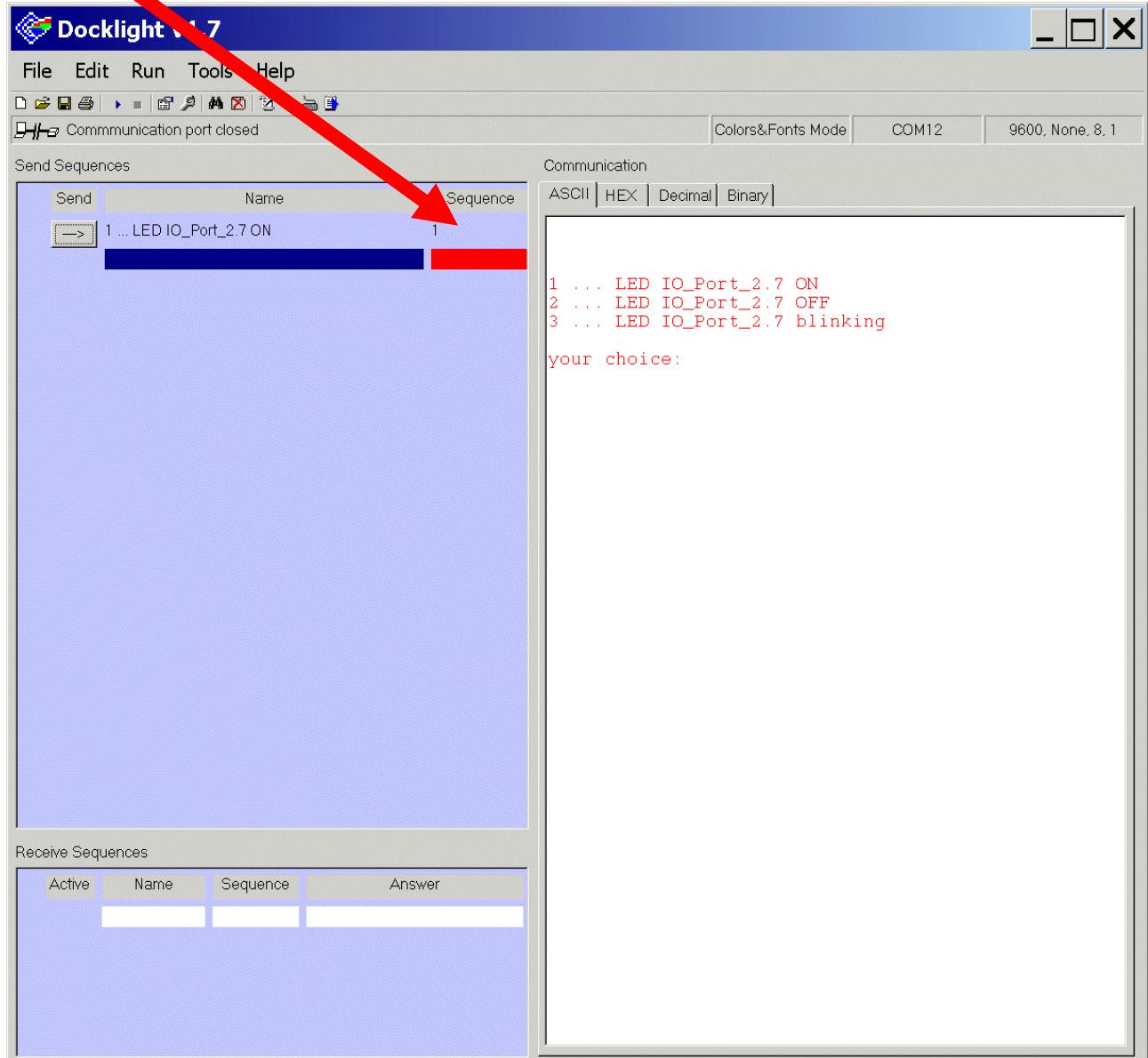


Edit Send Sequence: Sequence Definition: 1- Name: insert: 1 ... LED IO\_Port\_2.7 ON  
Edit Send Sequence: Sequence Definition: 2- Sequence: insert: 1

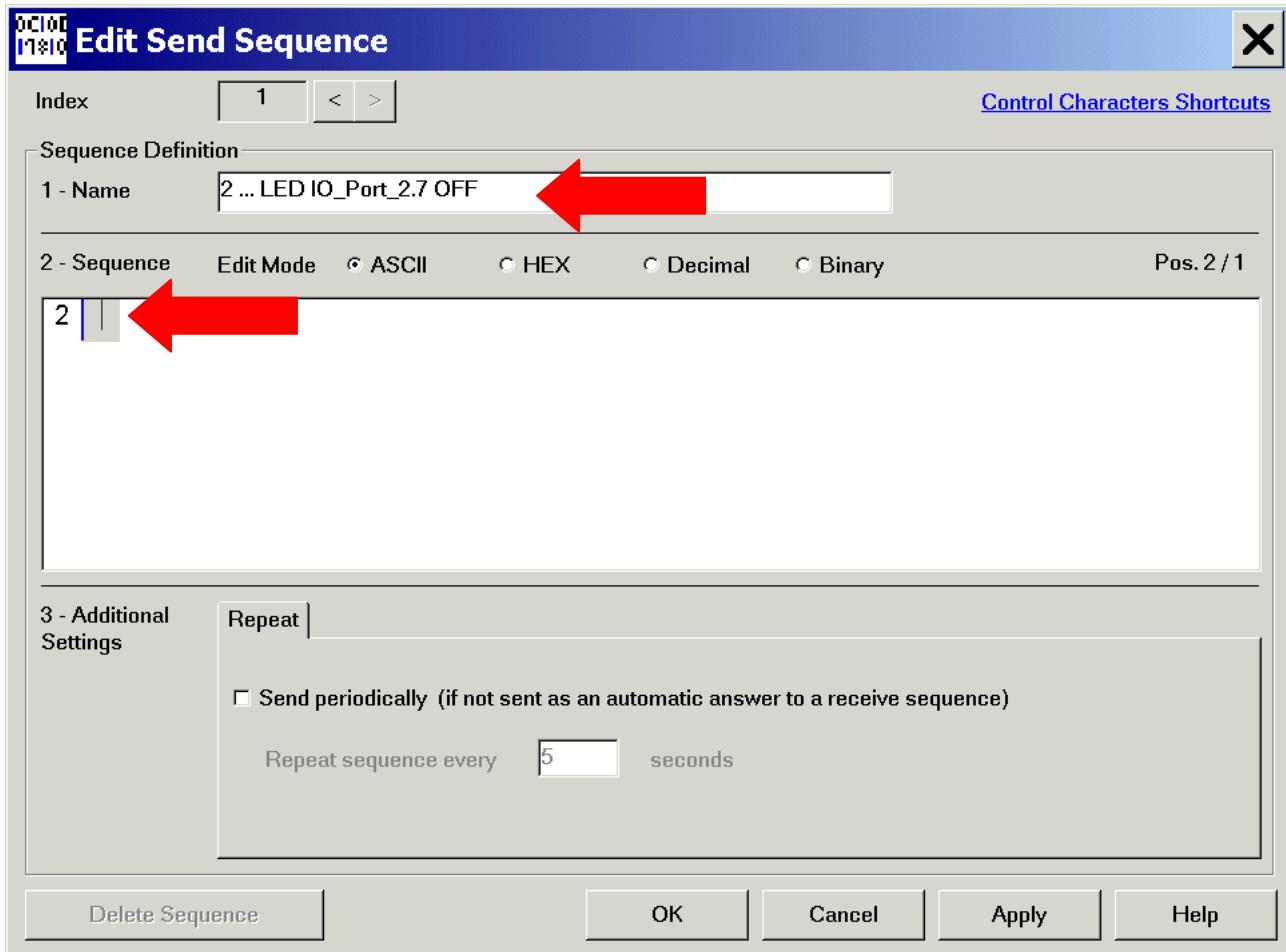


OK

Double click inside the red box:

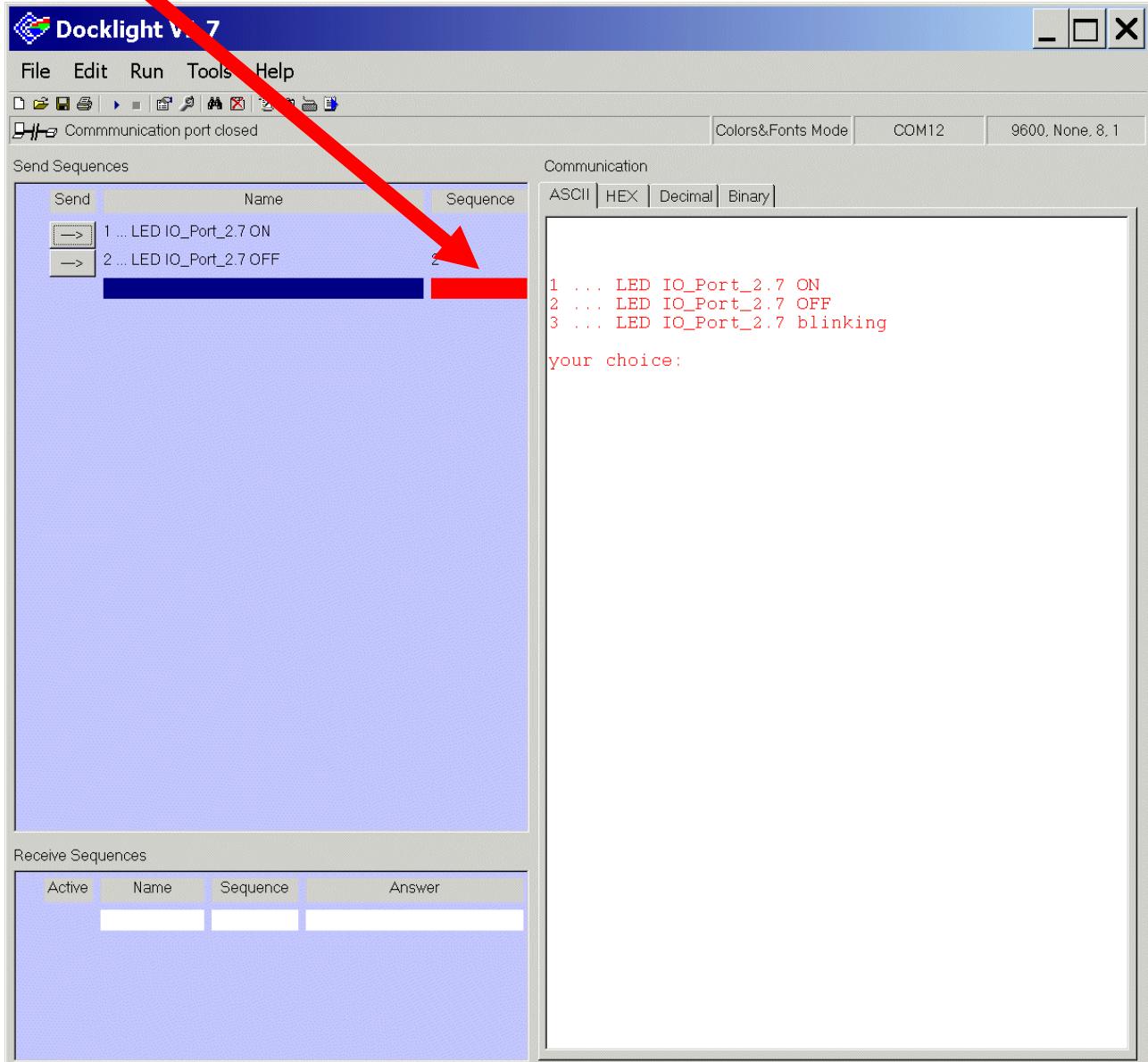


Edit Send Sequence: Sequence Definition: 1- Name: insert: 2 ... LED IO\_Port\_2.7 OFF  
Edit Send Sequence: Sequence Definition: 2- Sequence: insert: 2

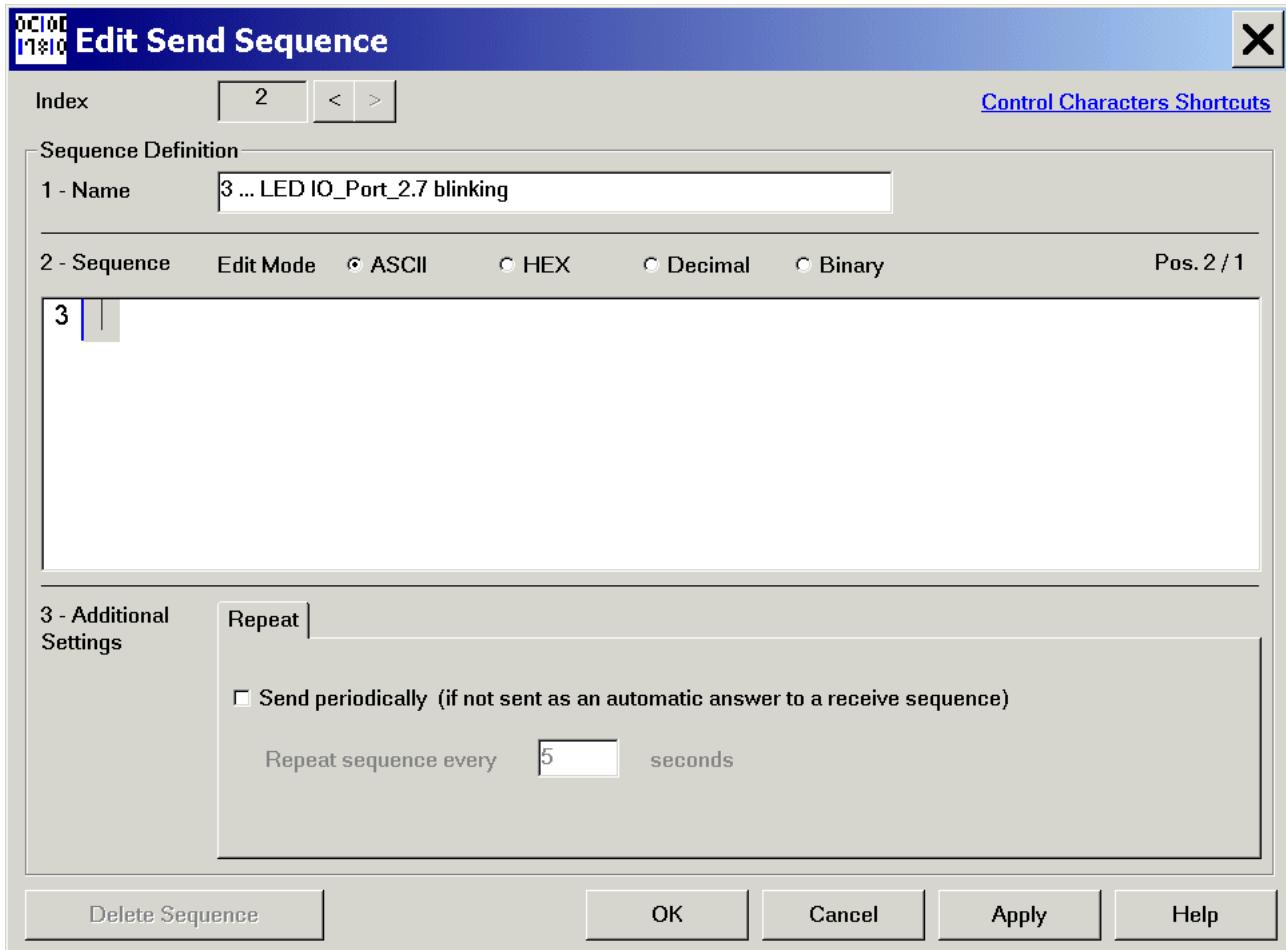


OK

Double click inside the red box:

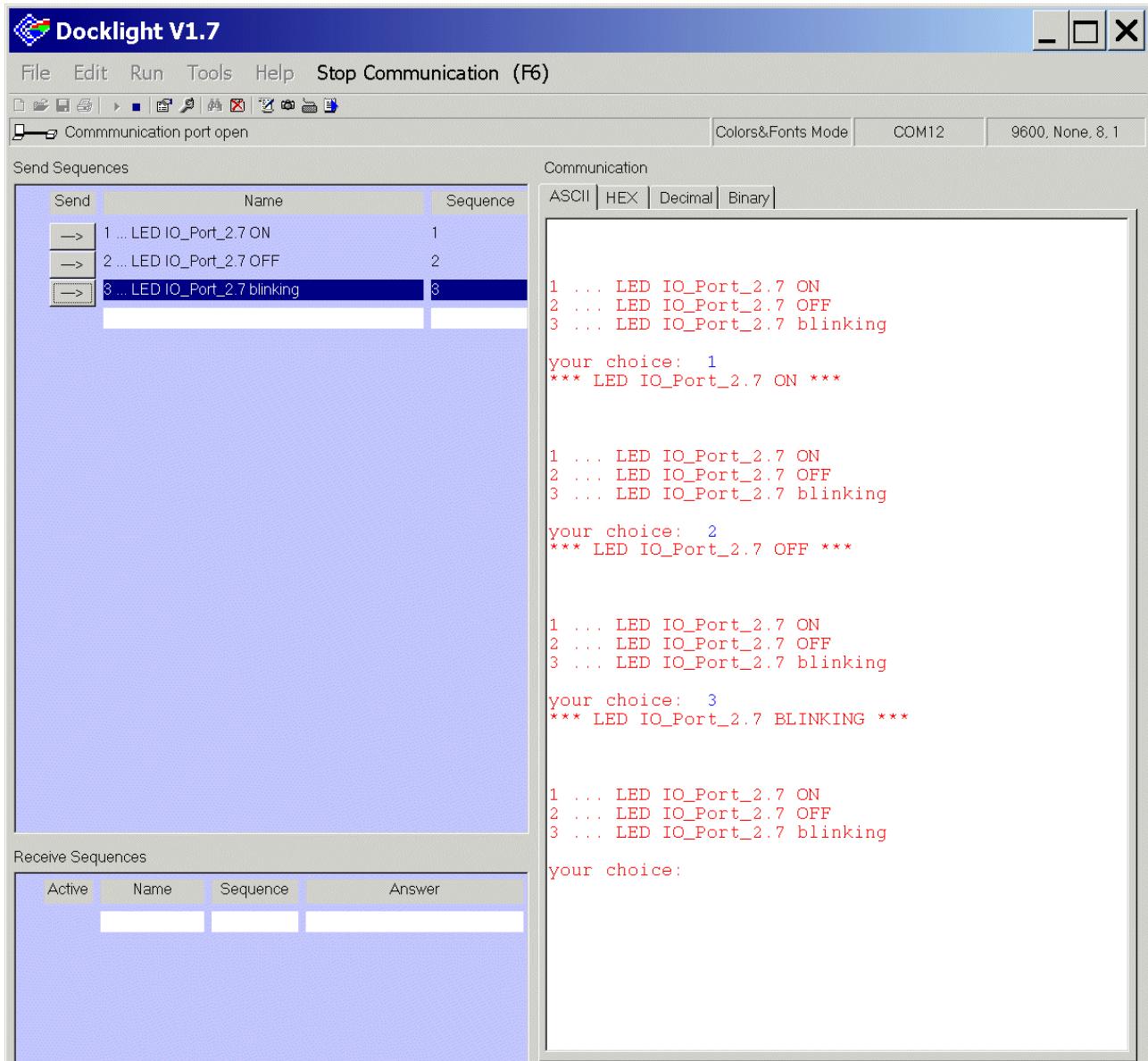


Edit Send Sequence: Sequence Definition: 1- Name: **insert: 3 ... LED IO\_Port\_2.7 blinking**  
Edit Send Sequence: Sequence Definition: 2- Sequence: **insert: 3**

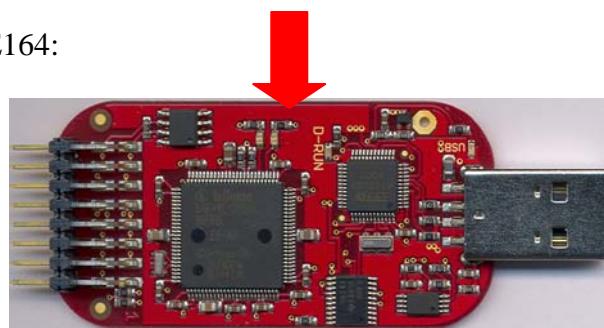


OK

Click 1 ... LED IO\_Port\_2.7 ON or  
 Click 2 ... LED IO\_Port\_2.7 OFF or  
 Click 3 ... LED IO\_Port\_2.7 blinking



and **check** the results on your UConnect-CAN XE164:



Now we close our project and µVision 4:

**Project - Close Project**

**File**  
**Exit**



### Conclusion:

In this step-by-step book you have learned how to use the UConnect-CAN XE164 together with the Keil tool chain.

Now you can easily expand your "hello world" program to suit your needs!

You can connect either a part of - or your entire application to the UConnect-CAN XE164.

You are also able to benchmark any of your algorithms to find out if the selected microcontroller fulfills all the required functions within the time frame needed.

Have fun and enjoy working with XE16x microcontrollers!

### Note:

There are step-by-step books for 8 bit microcontrollers (e.g. XC866, XC888 and XC878), 16 bit microcontrollers (e.g. C16x, XC16x and XE16x/XC2xxx) and 32 bit microcontrollers (e.g. TC1796 and TC1130).

All these step-by-step books use the same microcontroller resources and the same example code.

This means: configuration steps, function names and variable names are identical.

This should give you a good opportunity to get in touch with another Infineon microcontroller family or tool chain!

There are even more programming examples using the same style available [e.g. ADC-examples, CAPCOM6-examples (e.g. BLDC-Motor, playing music), Simulator-examples, C++ examples] based on these step-by-step books.

**6.) Feedback (UConnect-CAN XE164, Keil tools, µVision4):**  
**Your opinion, suggestions and/or criticisms**



**Contact Details (this section may remain blank should you wish to offer feedback anonymously):**

---

---

---

If you have any suggestions please send this sheet back to:

**email:** mcdocu.comments@infineon.com  
**FAX:** +43 (0) 4242 3020 5783



**Your suggestions:**

---

---

---

---

---

---

---

---

---

---

---

---

---

---

<http://www.infineon.com>